Proceedings of the Workshop on Innovative Approaches to Planning, Scheduling and Control (pp. 109–114). San Diego, CA: Morgan Kaufmann (1990).

Toward an Experimental Science of Planning

Pat Langley and **Mark Drummond***
AI Research Branch, Mail Stop 244–17
NASA Ames Research Center
Moffett Field, CA 94035 USA

Abstract

In this paper we outline an experimental method for the study of planning. We argue that experimentation should occupy a central role in planning research, identify some dependent measures of planning behavior, and note some independent variables that can influence this behavior. We also discuss some issues of experimental design and different stages that may occur in the development of an experimental science of planning.

1. Experimentation in Planning Research

Many sciences, such as physics and chemistry, attempt to integrate theory and experiment. For instance, theoretical physicists make predictions that are tested by experimental physicists, and when prediction and observation differ, the theory must be revised. Such cooperation between theoretician and experimentalist is a sign of a field's maturity, and it should be encouraged whenever possible.

At first glance, AI work on planning may appear inherently different from the natural sciences. Because researchers study artifacts over which they have complete control, one might think there is no need for experimentation and that formal analysis should suffice. But this view ignores the fact that all theories rely on assumptions that may or may not hold when applied to actual algorithms or real-world domains. Testing theoretical predictions through experiments lets one gather evidence in favor of correct assumptions, and it can point toward modifications when assumptions prove faulty. Long-term progress in planning will depend on such interaction between the theoretical and experimental paradigms.

Also, the complexity of most planning methods makes it difficult to move beyond worst-case analyses, suggesting experimentation as the only practical approach to obtaining average-case results. Thus, the field promises to have a significant empirical component for the foreseeable future. And unlike some empirical sciences, such as astronomy and sociology, planning is fortunate enough to have control over a wide range of factors, making experimentation easy and profitable.

In any science, the goal of experimentation is to better understand a class of behaviors and the conditions under which they occur. Ideally, this will lead to empirical laws that can aid the process of theory formation. In our field, the central behavior is planning, and the conditions involve the algorithm employed and the environment in which planning occurs. An implemented planning algorithm is necessary but not sufficient; one should also attempt to specify both when it operates well and the reasons for its behavior. Experimentation can provide evidence on both these issues.

As normally defined, an *experiment* is a study in which one systematically varies one or more *independent* variables and examines their effect on some *dependent* variables. Thus, a planning experiment involves more than running a planning algorithm on a single problem; it involves a number of runs carried out under different conditions. In each case, one must measure some aspect of planning behavior for comparison across the different conditions. Below we consider some dependent and independent variables that are relevant to planning research. We then turn to broader issues in designing experiments and in developing an experimental science of planning.[1]

2. Dependent Measures of Behavior

To evaluate any planning system, one needs some measures of its behavior. In most experiments, these are the dependent variables that one would like to predict. There are two obvious classes of metrics for planning algorithms – the *quality* of the generated plans and the *effort* required to generate them.

There exist many variations on the notion of plan quality. In a classical planning framework, one might

*Also affiliated with Sterling Federal Systems.

[1] For other discussions of experimentation in AI, see Kibler and Langley (1988) and Cohen and Howe (1988).

simply measure the length of the solution path or the total number of actions. More sophisticated dependent variables involve the time taken to execute a plan, the energy required, or the use of other resources. Alternatively, one can examine the robustness of a plan, as would be characterized by its ability to respond well under changing or uncertain conditions.

However, in many domains, finding any plan at all requires significant search, making it important to measure the time or effort spent in generating a plan. Measures of this sort have predominated in recent experimental studies of learning in planning domains (e.g., Minton, 1990; Iba, 1989). The simplest measure involves the total CPU time, but this metric can depend on both machines and implementations. More appropriate measures include the number of nodes considered in a search tree (Minton, 1990; Mooney, 1989), the number of unifications required (Allen & Langley, 1990), and the number of subgoals generated during the planning process (Jones, 1989). Of course, such internal measures are less interesting for intelligent agents that interact with an external environment; in such cases, measures of overall external time for planning and execution become relevant, despite possible differences in hardware.

Most measures of plan quality and planning effort implicitly assume that the planner will find a solution to every problem, but this is unrealistic in resource-limited situations. In such cases, the agent may be unable to solve certain problems, and it is important to take this into account when reporting experimental results. One response involves explicitly incorporating this result into the quality measure by giving unsuccessful attempts a very low score. Incorporating these cases into measures of effort is more difficult. As Segre, Elkan, and Russell (1990) have noted, averaging failed problems into effort scores can bias results in favor of one system over another. Alternatively, one can simply report the percentage of solved problems, treating this as a separate dependent measure.

3. Comparative Studies of Planning

Informal comparisons among planning algorithms abound in the AI literature, but there are relatively few systematic experiments that examine the behavior of different algorithms on the same problems. However, such comparative studies have an important role to play in developing a well-founded discipline.

3.1 Gross Comparisons of Planning Methods

The simplest form of planning experiment involves comparing the behavior of entirely different algorithms on the same problem or problems. In this case, the independent variable is the particular planning system being used and the dependent variable is one or more of the measures described above. For instance, Sacerdoti compared the behavior of a simple means-ends planner to that of a planner incorporating means-ends analysis and abstraction. More recently, Ruby and Datta (1990) have reported more extensive experiments, comparing these two approaches in terms of nodes searched and length of solution path. One can also imagine experimental comparisons between preplanning and reactive systems, between search-based and case-based methods, and between specific algorithms within the same basic paradigm.

In such comparative studies, it is important to place the systems' behavior in context. To this end, one can usually compare their performance to that of a 'straw man' that uses a simple-minded strategy (e.g., a traditional nonlinear planner) on the same set of problems. If one of the 'advanced' algorithms actually carries out more search or generates lower-quality solutions than this naive approach, this is a cause for concern. Lower bounds of this sort help calibrate the quality of system behavior.

3.2 Parametric Studies of Planning

Gross comparisons between different planning methods have the aura of a competition, in which one method wins and the others lose. However, a science of planning should aim not for simple-minded conclusions but for increased understanding. To this end, researchers should attempt to identify the *reasons* for success or failure on a problem or class of problems, attempting to generalize beyond a specific system and experiment.

This goal requires finer-grained studies of planning algorithms and their behavior. For instance, many systems contain a set of user-specified parameters, and in such cases one can experimentally determine the effect of the parameter settings on system behavior. A number of parameters suggest themselves:

- in preplanning systems, the maximum amount of resources devoted to generating a plan (e.g., limits on time, memory, or search);
- in reactive systems, the frequency at which the agent samples its environment;
- in combined systems, the ratio of deliberation to execution (Maes, in press; Sutton, 1990); and
- in knowledge-intensive systems, the bias toward modifying stored plans versus dynamically constructing new plans.

Ideally, behavior will be 'acceptable' within a wide range of parameter values, with the system's behavior varying slowly as a function of the settings. Hopefully, the same range of values will work across a variety of domains.

A related issue concerns the evaluation function or control scheme that a planning system uses to direct search. If the function contains parameters, then one can examine their relative importance through simple

parametric studies. However, one can also replace the entire control scheme with different ones in an attempt to find improved search methods. For instance, in a case-based system one might compare an existing similarity criterion for indexing knowledge with other approaches, such as Bayesian methods.

3.3 Lesion Studies of Planning Components

Some planning systems contain a number of independent components, and one can study the usefulness of each by removing it from the system. In such a 'lesion' experiment,[2] one runs the system with and without a given component, measuring the difference in performance. If a component does not aid the overall planning process, then it can be removed without undesirable consequences. Some obvious candidates for lesioning include:

- mechanisms for abstraction planning;
- methods for hierarchical planning;
- heuristics for identifying when to replan; and
- techniques for handling special forms of goals.

The above components focus on processes, but one can also imagine lesioning *knowledge* from a system. For example, some planning systems (Wilkins, 1988) incorporate constraints that may narrow the search or improve solution quality, but the influence of these constraints on behavior is an empirical question. Similarly, case-based planning systems draw upon a library of plans (Hammond, 1989) or plan components (Jones, 1989) in the construction of new plans, and one can determine the change in behavior as one adds or removes cases from memory.

One special case of lesion studies focuses on learning, and much of the recent experimental work on planning falls into this area. In this paradigm, one runs a planning system with and without a learning component, then examines differences in performance between the two variants. Allen and Langley (1990), Iba (1989), Minton (1990), Ruby and Kibler (1989), and Shavlik (1990) report evidence that a variety of learning components can improve the behavior of planning systems after sufficient experience in a given domain.

In some cases, researchers have also found negative results; both Iba (1989) and Minton (1990) have shown that naive learning methods can actually degrade planning performance in terms of search required to find solutions. However, rather than abandoning the use of learning methods, both used their results to identify the source of degradation and went on to develop learning methods that improve performance. This work provides an excellent role model for those interested in the experimental study of planning. Kibler and Langley (1988) discuss additional issues that arise in experiments with learning planners, as do Segre et al. (1990).

[2] This approach is common in neuroscience, where researchers excise a specific region of the brain to determine its effect on behavior.

4. Varying the Planning Domain

Seldom will one system always appear superior to another, and this leads naturally to the idea of identifying the conditions under which one approach has better performance than another. To study the effect of the environment on a planning system, one must vary the domain in which it operates. Natural domains, such as path planning for an autonomous vehicle or manipulator planning for an industrial robot arm, are the most obvious because they show real-world relevance. Also, successful runs on a number of different natural domains provide evidence of generality.

The simplest approach to this issue involves designing a set of 'benchmark problems'. To be scientifically useful, each benchmark problem should highlight certain problem attributes to help isolate planners' particular abilities. In addition, a realistic set of benchmark problems can help the scientific community explain its results in terms that can make a difference to those concerned with practical applications. These two goals – fostering scientific comparison and engineering development – place rather different constraints on a set of benchmark problems.

For the purposes of scientific comparison, one must be able to independently vary different task attributes. To achieve this, some benchmark problems should involve *artificial* domains. For situations that involve planning and execution, relevant attributes relate to the initial state specification, the goals, and the domain dynamics. For instance, one might consider the following sorts of task attributes:

- the length of the 'optimal' solution path (e.g., the number of actions in a block-stacking task);
- the effective branching factor (e.g., the number of actions considered for each plan step);
- the complexity of the environment (e.g., the number of obstacles in a navigation task);
- the amount of goal interaction in a planning task;
- the reliability of the domain (e.g., the probability that effectors will have the desired effect); and
- the rate of environmental change not due to the agent's actions.

However the list of task attributes is constructed, the set of representative problems should provide a complete coverage of the task attribute space. Complete coverage will let researchers choose problems from the set that highlight the system capabilities they seek to measure.

The set of task attributes and benchmark tasks should evolve concurrently.

Artificial domains are gaining acceptance with the planning community (e.g., Pollack & Ringuette, in press), since they let researchers systematically study planning behavior across a wide range of situations.[3] Another advantage of artificial domains is that they specify a variety of domain characteristics. In many cases, this lets one determine plans having *optimal* quality, thus establishing upper bounds on a planner's output. One can then compare the plans generated by actual algorithms against these upper bounds. If plan quality approaches this bound, one can also decide whether additional components or extra computation are worth minor improvements in this regard.

For engineering development and technology transfer purposes, tasks that include 'practical' difficulties will be more useful. Domains involving physical output devices such as robot arms and physical input devices such as limit switches will prove more useful in terms of validating particular systems. It is important to include problems in the evolving set of benchmarks that support such engineering evaluation, but discussion of such issues is beyond the scope of this paper.

5. Issues in Experimental Design

Basic experimental method suggests that researchers vary one independent term at a time while holding others constant. However, one can repeat this technique many times to achieve 'factorial' designs that measure dependent variables under all combinations of independent values. Full factorial designs are impractical when many independent variables are involved, but reduced experimental designs are also possible.

The advantage of combinatorial designs is that they let one go beyond the effects of isolated factors and detect *interactions* between independent variables. For instance, one might find that planning method A behaves better than method B in environment X, whereas B fares better than A in environment Y. Alternatively, one might find that two components of a planning method lead to synergy, or that the joint presence of two domain characteristics make planning especially difficult. We anticipate that many of the most interesting results in planning will have this form. The detection of such interactions does more than establish the conditions under which alternative methods should be used; it can also suggest hybrid algorithms.

Another issue in experimental design involves the use of sampling and statistical tests. In the natural sciences, one can never control all possible variables. As a result, researchers must collect multiple observations for each cell in their experimental design, average the resulting values, and use statistical techniques to ensure that conclusions about differences between cells are justified by the data. Although in principle one can control all the factors that influence a planning system, for practical reasons this will seldom be possible, and planning researchers should consider using them as well.

For instance, seldom can one test a planning system on all possible problems from a given domain. Thus, it makes sense to select a random sample, run the system on all problems in this set, and report the mean and variance on dependent measures of interest. In some situations, the effects of the independent variables will be large enough that formal significance tests are not necessary. In other cases, the variances may be sufficiently high that statistics should be invoked. And though exploratory studies are useful, researchers often design experiments with some hypotheses in mind, and whenever possible they should explicitly state and test these hypotheses. In all cases, the experimenter should use caution and common sense in designing his or her experiments and in interpreting the results.

6. An Imaginary Experimental Study

An imaginary example may clarify the nature of planning experiments. Suppose Dr. Calvin has developed a new planning algorithm, OUTSTRIPS, in response to limitations of earlier systems, say an inability to scale to complex problems. In this case, the hypothesis is that the new method will 'outstrip' other systems as task complexity increases. This suggests two independent variables – the algorithm employed and the problem difficulty.

At this point, Dr. Calvin must settle on some measures of difficulty. Rather than using the number of actions in optimal solutions, she favors a more sophisticated metric that incorporates the idea of goal interaction.[4] She also decides to study the systems' behaviors in multiple domains, say an idealized manipulation task like the blocks world and an idealized navigation task. Similar results in multiple domains will lend credence to her findings, so she includes this as a third independent variable.

Calvin must also identify the dependent measures she plans to use, and the explicit hypotheses she hopes to test. Naturally, she is interested in solution quality, which she will measure as the number of actions in the final plan, but she is even more interested in planning effort. Calvin has implemented OUTSTRIPS on her new positronic hardware, but she must run the comparison algorithms (including a straw man) on archaic silicon machines. Since all the systems involved in the study define their search spaces in a similar manner, she decides

[3] One can also view resource limitations (e.g., time or energy) as independent variables that affect task difficulty. Experimental studies of 'anytime' algorithms (Dean & Boddy, 1988) might examine the effect of planning time on quality of the resulting plans.

[4] Jones (1989) provides an initial approach to measuring goal interaction for means-ends systems.

to use the number of expanded nodes as her measure of effort.

In carrying out her experiment, the researcher must randomly select from problems at each level of difficulty, since the number of possible problems increases rapidly with difficulty. However, Calvin is careful to use the same test problems for each system. For each problem, she measures the various systems' search and plan quality, recording the mean and variance for each system-difficulty combination. She follows this procedure in each of the planning domains selected for study.

In this case, let us suppose that, for each domain, OutStrips requires more search than its competitors on simple tasks, but that it expands considerably fewer nodes on difficult problems, with the gap widening as the difficulty increases. These results constitute evidence in favor of the original hypothesis that OutStrips scales better than other methods. However, Calvin also notes that her system's plan quality is slightly worse than that for the more expensive algorithms. As expected, she also notes that all systems perform better than the straw man, except on the simplest problems.

In response to these findings, Calvin designs a lesion study in an attempt to identify the particular constraints used by OutStrips that lead to its superiority. To this end, she repeats the above experiment with lesioned versions of her algorithm, finding that some constraints greatly reduce planning effort, but that one of them is partly responsible for decrements in plan quality. As a result, Calvin has not only arrived at a deeper understanding of her system's success (and how its constraints might be transferred to other systems); she has also determined that deletion of one component actually produces a superior system with respect to plan quality. Of course, this is not the end of the story, for additional experiments by other researchers may identify conditions under which OutStrips fares poorly, suggesting ideas for even better algorithms.

7. Toward an Experimental Science

Different goals are appropriate for different stages of a developing experimental science. Although planning work remains in the early steps of this evolution, it is worthwhile considering the states that may arise on the path toward a mature scientific discipline.

In the initial stages, researchers should be satisfied with qualitative regularities that show one method as better than another under certain conditions, or that show one environmental factor as more devastating to a certain algorithm than another. Experimental evaluations should become the norm for published papers, with researchers comparing new algorithms against well-tested systems that act as 'straw men'. Parametric and lesion studies should examine the contributions of specific components, leading to improved algorithms that build on limitations identified earlier. Comparative studies that examine different algorithms on the same domains should proliferate, not to show one method superior to another, but to suggest directions for improvement. Online libraries of representative domains should encourage such comparisons.

Later stages of planning research should move beyond qualitative conclusions, using experimental studies to direct the search for quantitative laws that can actually predict performance on unobserved situations. In the longer term, results of this sort should lead to theoretical analyses that explain such effects at a deeper level, using average-case methods rather than worst-case assumptions. For instance, Segre et al. (1990) outline a simple mathematical model of search in planning, which they propose to use in analyzing experimental results. Other researchers should follow this lead, aiming for robust theories of planning algorithms that predict behavior in novel experimental situations. Failed predictions should lead in turn to revised theories, in the same fashion that experiment and prediction interact in the natural sciences.

In summary, the planning field has already started its development toward an experimental science, and future advances should produce improved dependent measures, better independent variables, more useful experimental designs, and ultimately an integration of theory and experiment. However, even the earliest qualitative stages of an empirical science can strongly influence the direction of research, identifying promising methods and revealing important roadblocks. Research on planning is just entering this first stage, but we believe the field will progress rapidly once it has started along the path of careful experimental evaluation.

Of course, the potential benefits of experimentation do not mean that empiricists should report gratuitous experiments any more than theoreticians should publish vacuous proofs. Whether they lead to positive or negative results, experiments are worthwhile only to the extent that they illuminate the nature of planning mechanisms and the reasons for their success or failure. Although experimental studies are not the only path to understanding, we feel they constitute one of planning's brightest hopes for rapid scientific progress.

Acknowledgements

We would like to thank John Allen for useful comments on an earlier draft. Parts of this paper are similar to an earlier manuscript on research in machine learning, co-authored with Dennis Kibler, who has greatly influenced our ideas on experimentation.

References

Allen, J., & Langley, P. (1990). *The acquisition, organization, and use of plan memory* (Technical Report). Moffett Field, CA: NASA Ames Research Center, AI Research Branch.

Cohen, P. R., & Howe, A. E. (1988). How evaluation guides AI research. *AI Magazine*, *9*, 35–43.

Dean, T., & Boddy, M. (1988). An analysis of timedependent planning. *Proceedings of the Seventh National Conference on Artificial Intelligence* (pp. 49–54). St. Paul, MN: Morgan Kaufmann.

Hammond, K. J. (1989). Case-based planning: Viewing planning as a memory task. In B. Chandrasekaran (Ed.), *Perspectives in artificial intelligence*. Boston: Academic Press.

Iba, G. A. (1989). A heuristic approach to the discovery of macro-operators. *Machine Learning*, *3*, 285–317.

Jones, R. (1989). *A model of retrieval in problem solving*. Doctoral dissertation, Department of Information & Computer Science, University of California, Irvine.

Kibler, D., & Langley, P. (1988). Machine learning as an experimental science. *Proceedings of the Third European Working Session on Learning* (pp. 81–92). Glasgow: Pittman.

Maes, P. (in press). How to do the right thing. *Connection Science*.

Minton, S. (1990). Quantitative results concerning the utility of explanation-based learning. *Artificial Intelligence*, *42*, 363–391.

Mooney, R. (1989). The effect of rule use on the utility of explanation-based learning. *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence* (pp. 725–730). Detroit, MI: Morgan Kaufmann.

Pollack, M. E., & Ringuette, M. (in press). Introducing the Tileworld: Experimentally evaluating agent architectures. *Proceedings of the Eighth National Conference on Artificial Intelligence*. Cambridge, MA: AAAI Press.

Ruby, D., & Datta, P. (1990). *Reacting to interactions in abstract plans*. Unpublished manuscript, Department of Information & Computer Science, University of California, Irvine.

Ruby, D., & Kibler, D. (1989). Learning subgoal sequences for planning. *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence* (pp. 609–614). Detroit, MI: Morgan Kaufmann.

Sacerdoti, E. D. (1974). Planning in a hierarchy of abstraction spaces. *Artificial Intelligence*, *5*, 115–135.

Segre, A., Elkan, C., & Russell, A. (1990). *On valid and invalid methodologies for experimental evaluations of EBL* (Technical Report 90–1126). Ithaca, NY: Cornell University, Department of Computer Science.

Shavlik, J. W. (1990). Acquiring recursive and iterative concepts with explanation-based learning. *Machine Learning*, *5*, 39–70.

Sutton, R. S. (1990). Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. *Proceedings of the Seventh International Conference on Machine Learning* (pp. 216–224). Austin, TX: Morgan Kaufmann.

Wilkins, D. E. (1988). *Practical planning: Extending the classical AI planning paradigm*. San Mateo, CA: Morgan Kaufmann.