

A Survey of Planning and Scheduling Research at the NASA Ames Research Center

Monte Zweben
 NASA Ames Research Center
 Moffett Field, CA 94035
 (415) 694-6940
 zweben@pluto.arc.nasa.gov

Abstract

NASA Ames Research Center has a diverse program in planning and scheduling. This paper highlights some of our research projects as well as some of our applications. Topics addressed include machine learning techniques, action representations and constraint-based scheduling systems. The applications discussed are planetary rovers, Hubble Space Telescope scheduling, and Pioneer Venus orbit scheduling.

1 Introduction

NASA Ames Research Center's Artificial Intelligence Research Branch, led by Dr. Peter Friedland, has a diverse research program in planning and scheduling. Our work ranges from state-of-art fundamental research to applications of both new and existing technology. This paper is intended to summarize and highlight some of these activities.

The research issues we will highlight include: machine learning and planning, planning representations, non-symbolic representations, constraint-based scheduling, and the representation of procedural knowledge.

The applications we will present include Hubble Space Telescope scheduling, Mars Rover planning and scheduling, and Pioneer Venus orbit scheduling.

2 Planning and Scheduling

It is important to clarify the terms "planning" and "scheduling" before we proceed. An agent *plans* by finding actions that will take it from its current state to another desired state. Classically, this is a goal directed search through a space of possible partial plans. *Scheduling*, on the other hand, refers to an agent placing explicit times or orderings on a set of intended actions. This is usually a search through a space of possible timelines. In short, we call the process of finding actions that achieve goals *planning* and we call the placement of times on those actions *scheduling*.

3 Research

Our research program is a mix of internal research, university grants, and commercial contracts. Here we will present a representative subset of the program conducted at Ames, SRI, Stanford, and Carnegie-Mellon.

3.1 Learning in Planning

One of our group's areas of focus is machine learning and we are particularly interested in its application to planning and scheduling. We are exploring ways to improve search performance through the application of explanation-based learning techniques [Mit87,DeJ87]. The main idea behind this work is that a system can improve its performance by analyzing the solutions to problems it has previously encountered. As a result of this analysis, the system can remember the good decisions it made

as well as the poor ones. Ideally, we would like the system to generalize from this analysis so that the knowledge gained from its retrospection will be useful in cases that are not only identical to the ones it encountered, but also those that are close enough so that the previous experience would prove relevant and helpful.

Dr. Steven Minton, of Carnegie-Mellon University, performed a thorough analysis of a planning and learning system called PRODIGY [Min87,Min88]. PRODIGY is a STRIPS-like planner that employs explanation-based learning to acquire search control knowledge. His results showed that learning will not necessarily improve the performance of a planning system and in many cases it can degrade performance. As a result, Dr. Minton explored various methods of monitoring the utility of learned knowledge in order to transform (or possibly remove) learned knowledge to make the overall system more useful. Dr. Minton has recently joined our laboratory and will continue exploring planning and learning issues.

Another project within our laboratory is also addressing the utility problem in planning systems that learn. Monte Zweben and collaborators at the MITRE Corporation are specifically addressing the utility problem caused by the complexity of learned knowledge [Zwe88b]. When a planning system needs to make a decision it must consider the generalized information that it has learned. This pattern-matching overhead can overwhelm the system to the point where learned knowledge no longer aids efficiency. Using PRODIGY as a model, Mr. Zweben and his colleagues are developing a system that employs explanation-based learning (EBL) to acquire search knowledge, but relaxes some of the constraints usually associated with EBL techniques. Specifically, EBL generalizes from a single instance and guarantees the correctness of the learned knowledge. As a result, the learned information tends to be quite complex. This project's main extension to the PRODIGY model is the approximation of learned knowledge in the interest of lowering the expense of the relevancy check. As a result, this approximation of learned knowledge could be incorrect and must be monitored. If the learned knowledge is approximated erroneously and misleads the planner frequently, then the approximations must be refined. The goal of this project is to determine the approximation and refinement strategies that will result

in an efficient and effective collection of knowledge learned by an explanation-based component.

3.2 Planning Representations

Dr. Mark Drummond, of our group, takes a Net Theory approach to the problem of planning, scheduling and control [Dru85,Dru87]. His approach has a number of interesting features and advantages. Similar to Amy Lansky's [Lan87] work, it views a plan as a set of constraints over a pre-specified set of actions. Unlike Lansky's GEM model, however, the Net Theory approach allows one to distinguish clearly between orderings required by causality, and those that are simply convenient, given the agent's goals. The Net Theory approach also begins to make clear the true role of *least commitment planning*, where orderings on actions are postponed until an ordering decision *must* be made. Current plan representations frequently over-commit to specific orderings. This over-commitment is critical when dealing with complicated scheduling problems, since many orderings and conditions cannot be determined until a schedule is actually being carried out. The Net Theory approach currently being explored by Dr. Drummond allows complete postponement of ordering decisions until all environmentally determined information is available. This permits a new view on the role of an agent's synthetic temporal data structure. These data structures can now be viewed as plans, schedules, or control programs, depending on the phase of overall system operation. This work does not view planning and scheduling as a one-time process, but rather, includes an explicit control phase where plans/schedules are incrementally modified to suit execution needs.

Dr. Drummond is also exploring a number of other issues in his planning research including: the tradeoff of reactive and predictive scheduling, the role of means-ends analysis in planning, the integration of planning and scheduling mechanisms, the representation and derivation of conditional and iterative plans, the role of constraint-satisfaction in the planning process, and the use of domain constraints to control planning search [Dru88].

3.3 Control Without Symbols

The work of Dr. Stan Rosenchein, formerly of SRI International and now of Teleos Research, takes the perspective that expensive symbolic processing at run time can be avoided by compiling symbolic representations into circuitry guaranteed to act in bounded time. Dr. Rosenchein and his colleague Leslie Kaelbling have developed a set of tools that enables one to design a robotic controller in a high-level language, which then gets compiled into efficient circuitry that can be simulated or manufactured in hardware [Kae88,Ros86]. The fundamental idea behind this work is that much of the expensive search (like pattern matching) employed by symbolic reasoners can be accomplished at compile time, allowing the robot to quickly process its sensory information and react appropriately. One of their tools, Gapps [Kae88], takes a goal expression and rules in a goal decomposition language and outputs circuitry that will enable a system to take action given a goal and its current state. Their tool REX allows one to specify behavior that takes sensory input and the system's current state and updates the current state to reflect what has occurred in the system's environment. REX allows one to specify the circuitry in a language more abstract than circuits, but less abstract than that of a programming language. They are currently designing a system called RULER which will allow one to design the state update circuitry in a logical language resembling PROLOG. Ultimately, this language will be compiled into REX specifications.

This work is distinguished in that the REX language has been specifically designed to support analysis of any particular REX program to prove its correctness. Further, this work is currently used to control Flakey, the SRI mobile robot. We view this work as a realistic first step towards the production of efficient robotic control tools. It begins to show how a designer can allocate computational resources at different phases of the design and deployment process.

3.4 Constraint-based Scheduling

As previously mentioned, scheduling is the process of placing a pre-specified set of actions on a timeline ensuring that the schedule's constraints are maintained. One of our projects, led by Monte Zweben, addresses the formulation and resolu-

tion of complex scheduling and resource allocation problems using constraints to represent scheduling knowledge and preferences [Zwe88a]. Constraints are declarative representations of relationships that abstract away control flow. They allow one to specify the relationships between the problem's variables in a system and enable the system to automatically determine the computation path from known variables to the unknown [Sta77]. These representations can be used for lookahead in a search process. Lookahead or constraint propagation results in less backtracking (i.e., fewer futile search paths) because commitments to various choices in the system are made only if they are compatible with the choices remaining in the system [Har80,Ste80]. However, lookahead can result in unnecessary constraint propagation. To circumvent this problem, we employ a technique called delayed evaluation [Fil84]. A system employing delayed evaluation does not completely evaluate its data structures until they are accessed. We use the data structure *streams* [Abe85] which are lists that delay the evaluation of their tails (i.e., all the elements of the list except the first element). The use of streams is advantageous for two main reasons: 1) their delayed evaluation circumvents unnecessary constraint propagation; 2) their delayed evaluation is transparent to knowledge engineers because stream operations are quite similar to list operations and our model of constraint-satisfaction is based upon list operations.

3.5 Procedural Knowledge

Dr. Michael Georgeff of SRI International has developed a system called PRS - Procedural Reasoning System - that enables one to represent and use complex procedural knowledge [Geo86]. PRS takes a set of procedures and executes them in a goal-directed manner. It uses a declarative representation of procedures that extends the expressiveness of previous action representations. Actions in PRS can exhibit iteration and recursion and also can employ run-time conditional branching. Thus, decisions as to what action to perform next can be dependent upon the runtime environment. PRS procedures can also be interrupted by other procedures, thereby allowing emergency recognition and exception handling. The ability to change its focus of attention quickly and to act conditionally makes PRS a highly reactive system.

PRS also has interesting theoretical aspects in that it meets much of the rational agency criteria proposed in the recent philosophical literature. Because PRS behaves like a rational agent there is potential for the development of interesting explanation components. PRS has been exercised in a very complex and interesting domain: malfunction handling for the reaction control system of the Space Shuttle. NASA diagnostic manuals were encoded in PRS resulting in an extensible set of semi-autonomous procedures.

4 Applications

The Ames AI Research Lab performs state-of-the-art research, but does so in the context of real-world applications. This allows us to both verify that our methods scale-up to real problems and focus our research towards topics of interest to NASA. In addition to framing our research within NASA problems, we also demonstrate the utility of known AI techniques with engineering applications. Don Rosenthal is the director of our applications work. His applications projects include Pioneer Venus satellite scheduling and Hubble Space Telescope scheduling. In fiscal year 1989, Mr. Rosenthal will explore planetary rover applications.

4.1 Pioneer Venus

This project, now completed, showed the utility of rule-based systems for operational software [Ros88]. We developed a heuristic ground-based scheduler for science operations (e.g., instrument configurations, data storage and playback, telemetry, etc.) onboard the Pioneer Venus satellite. This software is currently performing a task in minutes which formerly took people hours. Further, the resulting schedules are as effective as the man-made ones but contain fewer flaws. The satellite's operations are currently scheduled with this expert system. This scheduler is the first expert system installed in day to day use within a NASA mission operations environment.

4.2 Hubble Space Telescope (HST) Scheduling

Thousands of proposed observations for HST must be processed by the Space Telescope Science In-

stitute (STScI), on the Johns Hopkins University campus in Baltimore, to construct schedules for the science operations of the orbiting optical observatory. Current software is not flexible or extensible enough to meet the operational demands expected on the system and we are helping to provide knowledge-based solutions to this problem [Mil87].

The HST projects we support take a constraint-based approach to scheduling. Dr. Stephen Smith, of Carnegie-Mellon University, is applying research in factory scheduling [Fox83,Smi86] to the HST problem. This approach is well suited for over-constrained problems where a solution requires the relaxation of constraints.

Another project, at the STScI, is applying state-of-the-art constraint satisfaction techniques to the HST scheduling problem. Their goal is to produce a flexible and extensible scheduler that can dynamically react to anomalies and re-schedule accordingly. This work has resulted in a program called SPIKE, which uses piecewise constant functions to quantitatively represent the degree of constraint violation. Using these functions, SPIKE can efficiently combine constraints as well as judge the options it must choose.

4.3 Planetary Rovers

In the coming year we will begin performing extensive research into the planetary rover problem while concentrating on the science planning and scheduling issues. Using the Mars Rover domain as a model, we are interested in rovers that can autonomously plan and execute an appropriate set of scientific analyses for many different science goals. Further, we will explore techniques that dynamically discover interesting science opportunities, and attempt to replan the rover's actions to accommodate these new goals.

Additionally, we will address the integration of navigation planning and science planning which will require research in systems that negotiate for resources and time.

We will also explore machine learning techniques that can improve the overall rover system. First, we will explore techniques that improve a system's search performance. Second, we will address model refinement for rovers that begin with a rough and incomplete model of their environment. These techniques review a system's actions and remembers when they succeed and when they fail. They

also find discrepancies between a system's expectations and its observations and uses these discrepancies to refine the system's models.

5 Summary

This paper is intended to selectively introduce our research and to point out references to technical papers. Some of the areas currently addressed by our group but not discussed here are: 1) planning with incomplete models [Car87b,Car87a], 2) the use of truth-maintenance in planning [Mor86], and 3) communicating, cooperating agents [Nil87]. In the coming year, we plan to expand our efforts in multi-agent planning and constraint satisfaction. The overall goal of the program is to develop the technology for large-scale automation of space missions.

6 Acknowledgements

Thanks to Mark Drummond, Peter Friedland, Steve Minton, Don Rosenthal, and Haym Hirsh for their contributions to this survey.

References

- [Abe85] Abelson, H., and Sussman, G.J. *Structure and Interpretation of Computer Programs*. MIT Press, Cambridge, MA, 1985.
- [Car87a] Carbonell, J.C. and Gil, Y. Learning by Experimentation. In *Proceedings of the Fourth International Workshop on Machine Learning*, Morgan Kaufman, Palo Alto, CA, 1987.
- [Car87b] Carbonell, J.G., Mason, Matthew T., and Mitchell, Tom M. Machine Learning and Planning in Reactive Environments. In *NASA Ames AI Forum Proceedings*, 1987.
- [DeJ87] DeJong, G.F. and Mooney, R. Explanation-Based Generalization: An Alternative View. *Machine Learning*, 1(2), 1987.
- [Dru85] Drummond, M.E. Refining and extending the procedural net. In *IJCAI-9 Proceedings*, Morgan Kaufman, Palo Alto, CA, 1985.
- [Dru87] Drummond, Mark E. A Representation of Action and Belief for Automatic Planning Systems. In *Proceedings of the AAAI/CSLI 1986 Workshop on Reasoning about Actions and Plans*, Morgan Kaufman, Palo Alto, CA, 1987.
- [Dru88] Drummond, M.E., and Currie, K.W. Exploiting temporal coherence in non-linear plan construction. *Computational Intelligence Journal*, to appear, 1988.
- [Fil84] Filman, R.E., Friedman, D.P. *Coordinated Computing: Tools and Techniques for Distributed Software*. McGraw-Hill Book Company, New York, NY, 1984.
- [Fox83] Fox, Mark S. *Constraint-Directed Search: A Case Study of Job Shop Scheduling*. PhD thesis, Carnegie-Mellon University, 1983.
- [Geo86] Georgeff, M.P. and Lansky, A.L. Procedural Knowledge. *Proceedings of the IEEE*, Special Issue on Knowledge Representation, 1986.
- [Har80] Haralick, R.M., Elliot, G.L. Increasing Tree Efficiency for Constraint Satisfaction Problems. *Artificial Intelligence*, 14, 1980.
- [Kae88] Kaelbling, L.P. Goals As Parallel Program Specifications. In *AAAI-88*, Morgan Kaufman, Palo Alto, CA, 1988.
- [Lan87] Lansky, Amy L. A Representation of Parallel Activity Based on Events, Structure, and Causality. In *Proceedings of the 1986 Workshop on Reasoning about Actions and Plans*, Morgan Kaufman, Palo Alto, CA, 1987.
- [Mil87] Miller, G., Rosenthal, D., Cohen, W., and Johnston, M. Expert Systems Tools for Hubble Space Telescope Observation Scheduling. *Telematics and Informatics*, 4(4), 1987.

- [Min87] Minton, S. and Carbonell, J.G. Strategies for Learning Search Control Rules: An Explanation-based Approach. In *IJCAI-87 Proceedings*, Morgan Kaufman, Palo Alto, CA, 1987.
- [Min88] Minton, S. Empirical Results Concerning the Utility of Explanation-based Learning. In *AAAI-88 Proceedings*, 1988.
- [Mit87] Mitchell, T.M., Keller, R.M., and Kedar-Cabelli, S.T. Explanation-Based Generalisation: A unifying view. *Machine Learning*, 1(1), 1987.
- [Mor86] Morris, P.H. and Nado, B. Representing Actions with an Assumption-based Truth-Maintenance System. In *AAAI-86*, 1986.
- [Nil87] Nilsson, Nils J., Cohen, Phillip R., Rosenchein, Stanley J., and Fertig, Kenneth. Intelligent Communicating Agents. In *NASA Ames AI Forum Proceedings*, 1987.
- [Ros86] Rosenchein, S.J. and Kaelbling, L.P. The Synthesis of Digital Machines with Provable Epistemic Properties. In *Proceedings of the Conference on Theoretical Aspects of Reasoning and Knowledge*, Morgan Kaufman, Palo Alto, CA, 1986.
- [Ros88] Rosenthal, D.A. and Jackson, Robert W. Development of a Pioneer Venus Expert Scheduling System. In *Proceedings of the 26th Aerospace Sciences Meeting AIAA-88*, 1988.
- [Smi86] Smith, S.F., Fox, M.S., and Ow, P.S. Constructing and Maintaining Detailed Production Plans: Investigations into the development of Knowledge-based Factory Scheduling Systems. *AI Magazine*, 7(4), 1986.
- [Sta77] Stallman, R.M., Sussman, G.J. Forward Reasoning and Dependency-Directed Backtracking in a System for Computer-Aided Circuit Analysis. *Artificial Intelligence*, 9, 1977.
- [Ste80] Stefik, Mark. *Planning With Constraints*. PhD thesis, Stanford University, January 1980.
- [Zwe88a] Zweben, M., Eskey, M., and Rosenthal, D. Constraint Satisfaction with Delayed Evaluation. *Submitted to CACM*, 1988.
- [Zwe88b] Zweben, Monte and Chase, Melissa P. Improving Operationality with Approximate Heuristics. In *Proceedings of the 1988 Spring Symposia Series in Explanation-based Learning*, 1988.