Resource-Bounded Sensing and Planning in Robotic Systems

A position paper

Shlomo Zilberstein*
Computer Science Department
University of Massachusetts

Abstract

This position paper is concerned with the implications of limited computational resources and uncertainty on sensing and planning in robotic systems. To address the computational complexity of sensor interpretation and planning processes, we redefine their principal role. Following Agre and Chapman's plan-as-communication approach, sensing and planning are treated as computational processes that provide *information* to an *execution architecture* and thus improve the overall performance of the system. We argue that robots must be able to trade off the quality of this information against its computational costs. Anytime algorithms, whose quality of results improves gradually as computation time increases, provide useful performance components for time-critical sensing and planning in robotic systems. The paper describes some of our current research directions and open problems.

The role of sensing and planning

Sensing[1] and planning are computational processes aimed at helping an autonomous robot (or an "agent") act intelligently in the world. The goal of sensing and planning is to improve the performance of an agent by providing certain types of information. A similar idea, termed *plan-as-communication*, was introduced by Agre and Chapman (1990). Agre and Chapman contrasted this approach with the more traditional approaches that they term *plan-as-program*. The plan-as-program view has several disadvantages related to its computational complexity, inability to handle imprecise sensory data and uncertainty. It also requires that plans be too detailed and it fails to relate the plan to the concrete situation.

We share Agre and Chapman's view of traditional planning. Our approach is similar to their plan-as-communication idea, but we offer an additional important step. *Since sensing and planning provide information to the execution architecture, their execution should be managed based on the value of this information.* Moreover, since the quality of the information degrades over time, an agent should trade off the quality of the information against the computation time needed to produce it.

At the heart of any robotic system lies an *execution architecture* that interacts with the physical world by performing certain *external actions*. The goal of sensing and planning is to provide the execution architecture with information that can improve the selection of actions. We distinguish between the actions selected by the planner to construct the plan and the actions selected by the execution architecture which are the *actual* actions performed by the robot. The two processes may be completely different and may sometimes disagree. The performance improvement that is achieved by the sensing and planning processes is reflected by the system reaching more of its goals in less time.

The sensing and planning modules of a robot should be designed to support a particular *execution architecture*. Unfortunately, much of the work on sensing and planning within the AI community has been carried out with respect to either trivial or ill-specified execution architectures. In reality, planning must be accompanied by an execution architecture that can translate the plan into individual external actions, monitor their execution, recognize various types of failures, recover from "simple" failures, and otherwise rely on re-planning. Some recent attempts to design planning systems have recognized the central role of execution architecture. For example, the Entropy Reduction Engine (ERE) (Bresina and Drummond 1990) is designed to integrate planning and reaction for control of NASA's autonomous rover. The ERE architecture includes three components, reactor, projector and reducer, that have the capability to operate independently and perform their assigned task. Planning performed by the projector is provided as advice to the reactor in order to improve its performance. Similarly, the reductor provides the projector with search control advice.

The robotics community has a long tradition of integrating planning with reaction. The typical approach has been to introduce a planner on top of a reactive execution architecture. For example, SROMA (Xiaodong and Bekey 1988) is an adaptive scheduler for mechanical assembly

*Author's address: University of Massachusetts, Computer Science Department, LGRC, Box 34610, Amherst, MA 01003-4610. Email: shlomo@cs.umass.edu. Phone: (413) 545-4189. URL: http://anytime.cs.umass.edu/~shlomo.

[1]The term "Sensing" refers to the computational task of perceptual data interpretation rather then the mechanical process of sensing.

tasks that utilizes plans that are generated off-line. The run-time scheduler adapts planned actions to the available resources. The planner in this architecture generates a *complete* plan off-line which is then down loaded into a reactive component. The reactive component is allowed to modify action ordering or introduce minor deviations. A similar approach has been used by Fox and Kempf (1985) for robot assembly tasks and by Fraichard and Laugier (1991) for robot motion planning. What distinguishes our approach from those systems is its capability to control the quality of planning on-line.

The value of sensing and planning

Sensing and planning help agents to choose actions intelligently. The ultimate goal of an agent is not to derive accurate domain description, "optimal" plans, or even "correct" (when generated) plans, but rather to transform the physical environment into a desired state and thus perform a certain task. While traditional computational properties such as complexity, correctness, and completeness are very important, they do not fully represent the goal of sensing and planning. The outcome of sensing and planning should be treated as a potentially valuable piece of *information*. The value of this information depends on three factors:

1. The objective quality of the information. In sensing, quality reflects the accuracy of the domain description. In planning, quality reflects the efficiency of the solution to the problem defined by the initial planning conditions.
2. The time at which the sensing and planning information becomes available to the system and the extent of change that might have occurred in the domain.
3. The capability of the agent to interpret the information produced by the sensing and planning processes and to exploit it effectively.

Standard decision-theoretic techniques can be used to determine the value of planning both analytically and experimentally (Howard 1966; Zilberstein and Russell 1993; Zilberstein 1996). This approach decouples the value of planning and sensing from their absolute correctness. Moreover, approximate information may be more valuable than precise information, if it can be produced much faster and if the robot has the necessary recovery mechanisms to cope with erroneous information.

Sensing and planning as resource-bounded computation

The various factors that determine the value of a plan are not independent. In particular, the deliberation time required to improve the quality of a plan will normally degrade the overall utility of the agent. Similar considerations apply to the quality of the domain description generated by the sensing process. Hence, it is useful to analyze sensing and planning as resource-bounded activities and to develop planning systems that can trade off planning quality for deliberation time. Since in many domains the value of continued planning is context dependent, a utility maximizing approach must rely on constant, real-time monitoring of the planning process.

Dean and Boddy (1988) Horvitz (1989), Russell and Zilberstein (1991) and others have shown that anytime algorithms offer a simple means by which an agent can trade off decision quality for deliberation time. In addition, efficient techniques have been developed for composition and monitoring of systems that are composed of anytime algorithms (Zilberstein 1993). We have shown that these techniques are useful for constructing and controlling the sensing and planning components of autonomous systems (Zilberstein 1996).

Over the past several years, the AI community has been concerned with the capability of deliberative systems to operate robustly in real-time domains. Some successful system architectures that combine deliberative and reactive components include Guardian (Hayes-Roth *et al.* 1992), a medical monitoring application based on a blackboard (BB*) architecture; Phoenix (Howe, Hart and Cohen 1990), a real-time system to control fire fighting; PRS (Ingrand, Georgeff and Rao 1992), a general architecture for combining reaction and deliberation; and CIRCA (Musliner, Durfee and Shin 1995), an architecture that combines real-time operation with AI techniques by cooperation between two separate subsystems. While these systems have totally different structure and characteristics, they share the assumption that deliberative AI techniques are inherently unpredictable in terms of performance. Our work adds an important capability to project (and optimize) performance using probabilistic performance profiles (Zilberstein 1995; Zilberstein and Russell 1996).

Beyond episodic problem solving

Historically, planning has been closely associated with problem solving and similar search techniques have been widely used in both areas (Hendler, Tate and Drummond 1990). However in many real-world environments planning is much more than traditional problem solving. Besides the richer representation of operators and goals in planning, a planning problem can contain much more than a single problem solving episode. The goal of a system may be "collect all the empty cans in the building" or "keep track of all the targets in region Z". When operating in a dynamic environment, translating such a high level goal into action may not have a fixed solution that can be derived and implemented. Achieving such goals requires an ongoing situation assessment, planning, and action. The problem of interleaving planning and execution is complex because it is hard to factor the importance of long term goals while executing a short term plan. The problem becomes much harder when the agent is operating under resource constraints.

Research directions and open problems

To improve current planning technology and make it more applicable to real-world situations, the following problems must be addressed:

1. How to formalize and analyze the relationship between the planner, the execution architecture, and the environment? Within our framework, the planner functions as a resource-bounded deliberation process that provides useful *information* to the execution architecture. This information is in the form of abstract operators that can guide the operation of the system.
2. How to develop planning algorithms that exhibit graceful degradation under resource constraints and that can operate concurrently with the system's execution architecture? This can be achieved using anytime planning techniques that iteratively refine a segment of an abstract plan.
3. How to develop a run-time meta-level resource manager that can monitor the execution of the planner so as to maximize its contribution to the overall performance of the system? How to perform cost effective monitoring of the deliberation process and the environment? Decision-theoretic techniques can be used to solve this deliberation scheduling problem using off-line compilation of performance profiles and on-line monitoring.
4. How to measure performance and evaluate planners when plans are used as advice to the execution architecture? Planners must be evaluated in the context of a complete working system and with respect to a particular set of environments. Performance evaluation should not end with summarizing the performance of each module. When the performance of the components is represented using conditional performance profiles, it can be used to introduce and exploit run-time tradeoffs between computational resources and overall performance.

Acknowledgments

Support for this work was provided in part by the National Science Foundation under grant number IRI-9409827 and by Rome Laboratory, U.S. Air Force, under grant number F30602-95-1-0012.

References

Agre, P. E. and Chapman, D. 1990. What are plans for? *Robotics and Autonomous Systems*, 6:17–34.

Bresina, J. and Drummond, M. 1990. Integrating planning and reaction. In J. Hendler, Ed., *AAAI Spring Symposium on Planning in Uncertain, Unpredictable or Changing Environments*, Stanford, California, pp. 24–28.

Dean, T. L. and Boddy, M. 1988. An analysis of time-dependent planning. In *Proc. 7th National Conference on Artificial Intelligence*, Minneapolis, Minnesota, pp. 49–54.

Fox, B. R. and Kempf, K. G. 1985. Opportunistic scheduling for robotic assembly. In *IEEE International Conference on Robotics and Automation*, pp. 880–889.

Fraichard, T. and Laugier, C. 1991. On-line reactive planning for a non-holonomic mobile in a dynamic world. In *IEEE International Conference on Robotics and Automation*, Sacramento, California, pp. 432–437.

Gat, E. 1992. Integration planning and reacting in a heterogeneous asynchronous architecture for controlling real-world mobile robots. In *Proc. 10th National Conference on Artificial Intelligence*, San Jose, California, pp. 809–815.

Hansen, E. A. and Zilberstein, S. 1995. Monitoring the progress of anytime algorithms. In *Proc. IJCAI-95 Workshop on Anytime Algorithms and Deliberation Scheduling*, Montreal, Canada, pp. 28–32.

Hayes-Roth, B. *et al.* 1992. Guardian: A prototype intelligent agent for intensive-care monitoring. *Artificial Intelligence in Medicine*, 4:165–185.

Hendler, J., Tate, A. and Drummond, M. 1990. AI planning: Systems and techniques. *AI Magazine*, 11(2):61–77.

Howard, R. A. 1966. Information value theory. *IEEE Transactions on Systems Science and Cybernetics*, SSC-2(1):22–26.

Howe, A. E., Hart, D. M. and Cohen, P. R. 1990. Addressing real-time constraints in the design of autonomous agents. In the *Journal of Real-Time Systems*, 2(1/2):81–97.

Horvitz, E. J. 1989. Reasoning about beliefs and actions under computational resource constraints. In *Uncertainty in Artificial Intelligence*, Vol. 3, Eds. L.N. Kanal, T.S. Levitt and J.F. Lemmer, pp. 301–324, North-Holland: Amsterdam.

Ingrand, F. F. Georgeff, M. P. and Rao, A. S. 1992 An architecture for real-time reasoning and system control. *IEEE Expert*, December 1992, pp. 34–44.

Musliner, D. J., Durfee, E. H. and Shin, K. G. 1995. World modeling for the dynamic construction of real-time control plans. *Artificial Intelligence*, 74:83–127.

Russell, S. J. and Zilberstein, S. 1991. Composing real-time systems. In *Proc. 12th International Joint Conference on Artificial Intelligence*, Sydney, Australia, pp. 212–217.

Xiaodong, X. and Bekey, G. A. 1988. Sroma: An adaptive scheduler for robotic assembly systems. In *IEEE International Conference on Robotics and Automation*, Philadelphia, Penn, pp. 1282–1287.

Zilberstein, S. 1993. *Operational Rationality Through Compilation of Anytime Algorithms*. Ph.D Dissertation, Computer Science Division, University of California, Berkeley, CA.

Zilberstein, S. 1995. Optimizing decision quality with contract algorithms. In *Proc. 14th International Joint Conference on Artificial Intelligence*, Montreal, Canada, pp. 1576–1582.

Zilberstein, S. 1996. Resource-bounded sensing and planning in autonomous systems. To appear in *Autonomous Robots*.

Zilberstein, S. and Russell, S. J. 1993. Anytime sensing, planning and action: A practical model for robot control. In *Proc. 13th International Joint Conference on Artificial Intelligence*, Chambery, France, pp. 1402–1407.

Zilberstein, S. and Russell, S. J. 1996. Optimal composition of real-time systems. To appear in *Artificial Intelligence*.