This quick précis of symbolic planning does not address many "classical" topics, such as learning or remembering useful plans. Also not discussed are: planning at varying levels of abstraction, plans with uncertain information, or plans with costs. The interested reader should consult the References for more information. The next section addresses plans with costs since they are particularly relevant to vision; some of the other issues appear in the Exercises.

13.2 PLANNING WITH COSTS

Decision making under uncertainty is an important topic in its own right, being of interest to policymakers and managers [Raiffa 1968]. Analytic techniques that can derive the strategy with the "optimal expected outcome" or "maximal expected utility" can be based on Bayesian models of probability.

In [Feldman and Sproull 1977] these techniques are explored in the context of action planning for real-world actions and vision. As an example of the techniques, they are used to model an extended version of the "monkey and bananas" problem of the last section, with multiple boxes but without the maddening pulley arrangement. In the extended problem, there are boxes of different weights which may or may not support the monkey, and he can apply tests (e.g., vision) at some cost to determine whether they are usable. Pushing weighted boxes costs some effort, and the gratification of eating the bananas is "worth" only some finite amount of effort. This extended set of considerations is more like everyday decision making in the number of factors that need balancing, in the uncertainty inherent in the universe, and in the richness of applicable tests. In fact, one might make the claim that human beings always "maximize their expected utility," and if one knew a person's utility functions, his behavior would become predictable. The more intuitive claim that humans beings plan only as far as "sufficient expected utility" can be cast as a maximization operation with nonzero "cost of planning."

The sequential decision-making model of planning with the goal of maximizing the goodness of the expected outcome was used in a travel planner [Sproull 1977]. Knowledge of schedules and costs of various modes of transportation and the attendant risks could be combined with personal prejudices and preferences to produce an itinerary with the maximum expected utility. If unexpected situations (canceled flights, say) arose *en route,* replanning could be initiated; this incremental plan ramification is a natural extension of sequential decision making.

This section is concerned with measuring the expected performance of plans using a single number. Although one might expect one number to be inadequate, the central theorem of decision theory [DeGroot 1970] shows essentially that one number is enough. Using a numerical measure of goodness allows comparisons between normally incomparable concepts to be made easily. Quite frequently numerical scores are directly relevant to the issues at stake in planning, so they are not obnoxiously reductionistic. Decision theory can also help in the process of applying a plan—the basic plan may be simple, but its application to the world may be complex, in terms of when to declare a result established or an action unsuccessful. The decision-theoretic approach has been used in several artificial intelligence and vision programs [Feldman and Yakimovsky 1974; Bolles 1977; Garvey 1976; Ballard 1978; Sproull 1977].

13.2.1 Planning, Scoring, and Their Interaction

For didactic purposes, the processes of plan generation and plan scoring are considered separately. In fact, these processes may cooperate more or less intimately. The planner produces "sequences" of *actions* for evaluation by the scorer. Each action (computation, information gathering, performing a real-world action) has a *cost*, expressing expenditure of resources, or associated unhappiness. An action has a set of possible *outcomes*, of which only one will really occur when the action is performed. A *goal* is a state of the world with an associated "happiness" or *utility*. For the purposes of uniformity and formal manipulation, goals are treated as (null) actions with no outcomes, and negative utilities are used to express costs. Then the plan has only actions in it; they may be arranged in a strict sequence, or be in loops, be conditional on outcomes of other actions, and so forth.

The *scoring* process evalutes the *expected utility* of a plan. In an uncertain world, a plan prior to execution has only an expected goodness—something might go wrong. Such a scoring process typically is not of interest to those who would use planners to solve puzzles or do proofs; what is interesting is the result, not the effort. But plans that are "optimal" in some sense are decidedly of interest in real-world decision making. In a vision context, plans are usually useful only if they can be evaluated for efficiency and efficacy.

Scoring can take place on "complete" plans, but it can also be used to guide plan generation. The usual artificial intelligence problem-solving techniques of progressive deepening search and branch-and-bound pruning may be applied to planning if scoring happens as the plan is generated [Nilsson 1980]. Scoring can be used to assess the cost of planning and to monitor planning horizons (how far ahead to look and how detailed to make the plan). Scoring will penalize plans that loop without producing results. Plan improvements, such as replanning upon failure, can be assessed with scores, and the contribution of additional steps (say for extra information gathering) can be assessed dynamically by scoring. Scoring can be arbitrarily complex utility functions, thus reflecting such concepts as "risk aversion" and nonlinear value of resources [Raiffa 1968].

13.2.2 Scoring Simple Plans

Scoring and an Example

A *simple plan* is a tree of nodes (there are no loops). The nodes represent actions (and goals). Outcomes are represented by labeled arcs in the tree. A probability of occurrence is associated with each possible outcome; since exactly one outcome actually occurs per action, the probabilities for the possible outcomes of any action sum to unity.

The *score* of a plan is its *expected utility*. The expected utility of any node is recursively defined as its utility times the probability of reaching that node in the plan, plus the expected utilities of the actions at its (possible) outcomes. The probability of reaching any "goal state" in the plan is the product of probabilities of outcomes forming a path from the root of the plan to the goal state.

As an example, consider the plan shown in Fig. 13.4. If the plan of Fig. 13.4

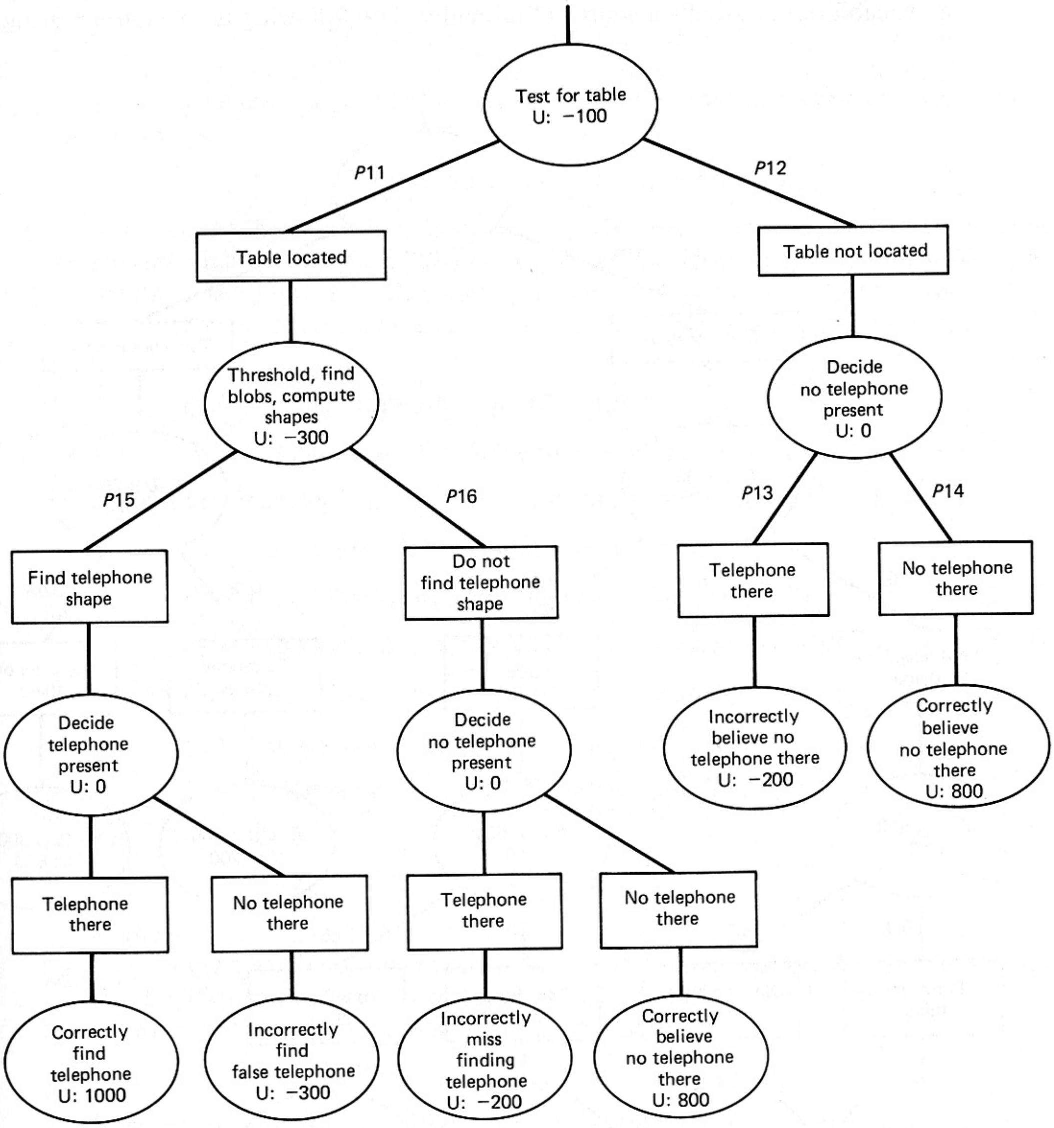


Fig. 13.4 This plan to find a telephone in an office scene involves finding a table first and looking there in more detail. The actions and outcomes are shown. The probabilities of outcomes are assigned symbols (P10, etc.). Utilities (denoted by U:) are given for the individual actions. Note that negative utilities may be considered costs. In this example, decision-making takes no effort, image processing costs vary, and there are various penalties and rewards for correct and incorrect finding of the telephone.

has probabilities assigned to its outcomes, we may compute its expected utility. Figure 13.5 shows the calculation. The probability of correctly finding the telephone is 0.34, and the expected utility of the plan is 433.

Although the generation of a plan may not be easy, scoring a plan is a trivial exercise once the probabilities and utilities are known. In practice, the assignment of probabilities is usually a source of difficulty. The following is an example using

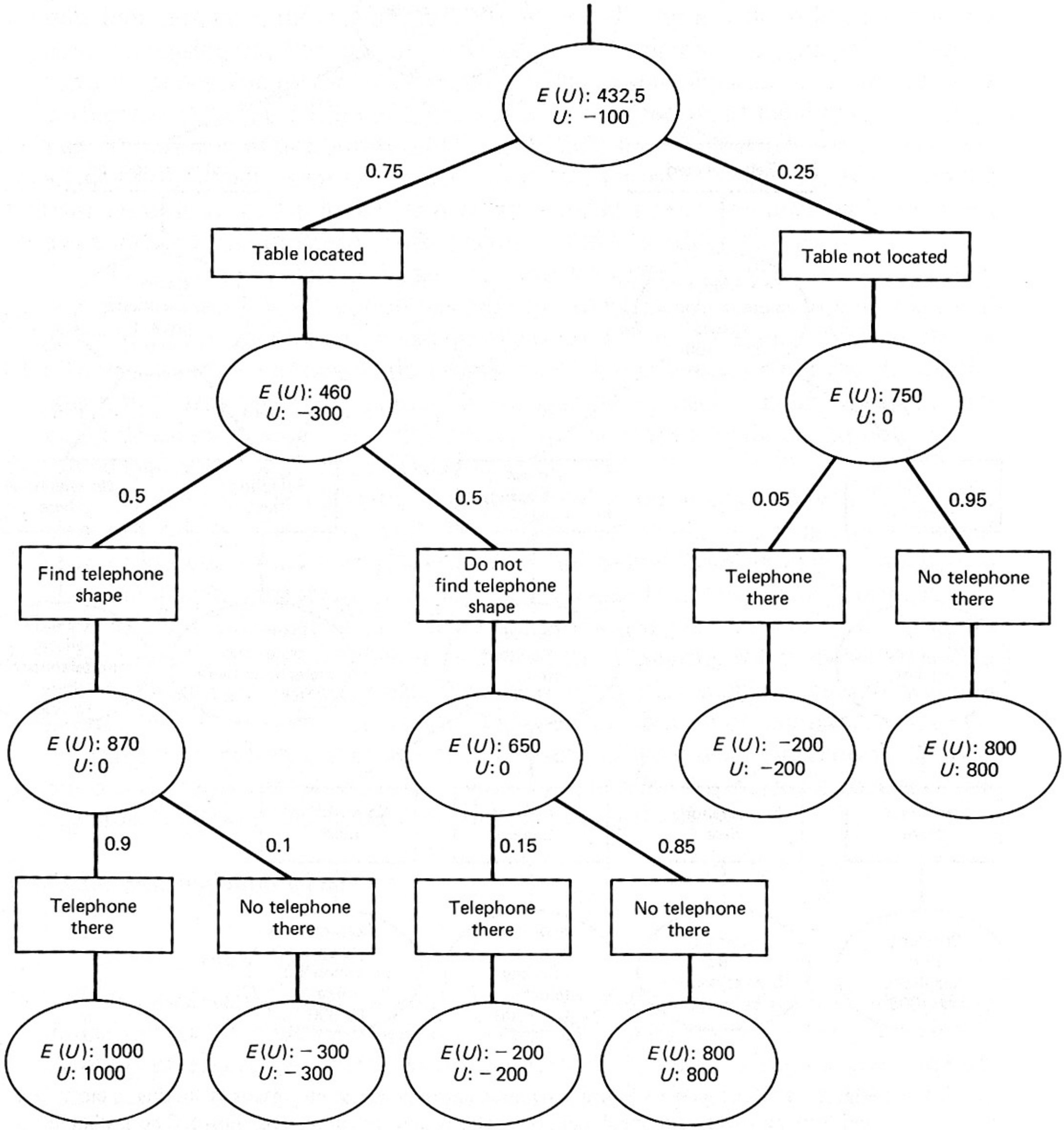


Fig. 13.5 As for Fig. 13.4. U: gives the utility of each action. E(U): gives the expected utillity of the action, which depends on the outcomes below it. Values for outcome probabilities are given on the outcome arcs.

the telephone-finding plan and some assumptions about the tests. Different assumptions yield different scores.

Computing Outcome Probabilities: An Example

This example relies heavily on Bayes' rule:

$$P(B|A)P(A) = P(A \wedge B) = P(A|B)P(B). \tag{13.1}$$

Let us assume a specific a priori probability that the scene contains a telephone.

$$P_1 = \text{apriori probability of Telephone} \tag{13.2}$$

Also assume that something is known about the behavior of the various tests in the presence of what they are looking for. This knowledge may accrue from experiments to see how often the table test found tables when telephones (or tables) were and were not present. Let us assume that the following are known probabilities.

$$P_3 = \text{P(table located|telephone in scene)} \tag{13.3}$$

$$P_5 = \text{P(table located|no telephone in scene)} \tag{13.4}$$

Either there is a telephone or there is not, and a table is located or it is not, so

$$P_2 = \text{a priori probability of no telephone} = 1 - P_1 \tag{13.5}$$

$$P_4 = \text{P(no table located|telephone in scene)} = 1 - P_3 \tag{13.6}$$

$$P_6 = \text{P(no table located|no telephone in scene)} = 1 - P_5 \tag{13.7}$$

Similarly with the "shape test" for telephones: assume probabilities

$$P_7 = \text{P(telephone shape located|telephone)} \tag{13.8}$$

$$P_9 = \text{P(telephone shape located|no telephone)} \tag{13.9}$$

with

$$P_8 = 1 - P_7, \qquad P_{10} = 1 - P_9 \tag{13.10}$$

as above.

There are a few points to make: First, it is not necessary to know exactly these probabilities in order to score the plan; one could use related probabilities and Bayes' rule. Other useful probabilities are of the form

$$\text{P(telephone|telephone shape located)}.$$

In some systems [Garvey 1976] these are assumed to be available directly. This section shows how to derive them from known conditional probabilities that describe the behavior of detectors given certain scene phenomena.

Second, notice the assumption that although both the outcome of the table test and the shape test depend on the presence of telephones, they are taken to be independent of each other. That is, having found a table tells us nothing about the likelihood of finding a telephone shape. Independence assumptions such as this are useful to limit computations and data gathering, but can be somewhat unrealistic. To account for the dependence, one would have to measure such quantities as

$$\mathrm{P}(\text{telephone shape found} \mid \text{table located}).$$

Now to compute some outcome probabilities: Consider the probability

$$P_{11} = \mathrm{P}(\text{table located}) \tag{13.11}$$

Let us write

TL for Table Located
TNL for Table Not Located.

A table may be located whether or not a telephone is in the scene. In terms of known probabilities, Bayes' rule yields

$$P_{11} = P_3\, P_1 + P_5\, P_2 \tag{13.12}$$

Then

$$P_{12} = P(\mathrm{TNL}) = 1 - P_{11} \tag{13.13}$$

Calculating P_{13} shows a neat trick using Bayes' Rule:

$$P_{13} = P(\text{telephone} \mid \mathrm{TNL}) \tag{13.14}$$

That is, P_{13} is the probability that there is a telephone in the scene given that search for a table was unsuccessful. This probability is not known directly, but

$$\begin{aligned} P_{13} &= \frac{P(\text{telephone and TNL})}{P(\mathrm{TNL})} \\ &= \frac{P(\text{TNL and telephone})}{P_{12}} \\ &= \frac{[P(\mathrm{TNL} \mid \text{telephone})\mathrm{P}(\text{telephone})]}{P_{12}} \\ &= \frac{[P_4\, P_1]}{P_{12}} \end{aligned} \tag{13.15}$$

Then, of course

$$P_{14} = 1 - P_{13} \tag{13.16}$$

Reasoning in this way using the conditional probabilities and assumptions about their independence allows the completion of the calculation of outcome probabilities (see the Exercises). One possibly confusing point occurs in calculation of P_{15}, which is

$$P_{15} = P(\text{telephone shape found} \mid \text{table located}) \tag{13.17}$$

By assumption, these events are only indirectly related. By the simplifying assumptions of independence, the shape operator and the table operator are independent in their operation. (Such assumptions might be false if they used common image processing subroutines, for example.) Of course, the probability of success of each depends on the presence of a telephone in the scene. Therefore their performance is linked in the following way (see the Exercises). (Write TSL for Telephone Shape Located.)

$$P_{15} = P(\text{TSL}|\text{TL})\,P(\text{TSL}|\text{telephone})\,P(\text{telephone}|\text{TL}) + P(\text{TSL}|\text{no telephone})\,P(\text{no telephone}|\text{TL}) \tag{13.18}$$

13.2.3 Scoring Enhanced Plans

The plans of Section 13.2.2 were called "simple" because of their tree structure, complete ordering of actions, and the simple actions of their nodes. With a richer output from the symbolic planner, the plans may have different structure. For example, there may be *OR* nodes, any one of whose sons will achieve the action at the node; *AND* nodes, all of which must be satisfied (in any order) for the action to be satisfactorily completed; *SEQUENCE* nodes, which specify a set of actions and a particular order in which to achieve them. The plan may have loops, shared subgoal structure, or goals that depend on each other. How enhanced plans are interpreted and executed depends on the scoring algorithms, the possibilities of parallel execution, whether execution and scoring are interleaved, and so forth. This treatment ignores parallelism and limits discussion to expanding enhanced plans into simple ones.

It should be clear how to go about converting many of these enhanced plans to simple plans. For instance, sequence nodes simply go to a unique path of actions. Alternatively, depending on assumptions about outcomes of such actions (say whether they can fail), they may be coalesced into one action, as was the "threshold, find blobs, and compute shapes" action in the telephone-finding plan.

Rather more interesting are the OR and AND nodes, the order of whose subgoals is unspecified. Each such node yields many simple plans, depending on the order in which the subgoals are attacked. One way to score such a plan is to generate all possible simple plans and score each one, but perhaps it is possible to do better. For example, loops and mutual dependencies in plans can be dealt with in various ways. A loop can be analyzed to make sure that it contains an exit (such as a branch of an OR node that can be executed). One can make ad hoc assumptions that the cost of execution is always more than the cost of planning [Garvey 1976], and score the loop by its executable branch. Another idea is to plan incrementally with a finite horizon, expanding the plan through some progressive deepening, heuristic search, or pruning strategy. The accumulated cost of going around a loop will soon remove it from further consideration.

Recall (Figs. 13.4 and 13.5) that the expected utility of a plan was defined as the sum of the utility of each leaf node times the probability of reaching that node. However, the utilities need not combine linearly in scoring. Different monotonic functions of utility express such different conceptions as "aversion to risk" or "gambling addiction." These considerations are real ones, and nonlinear utilities are the rule rather than the exception. For instance, the value of money is notoriously nonlinear. Many people would pay $5 for an even chance to win $15; not so many people would pay $5,000 for an even chance to win $15,000.

One common way to compute scores based on utilities is the "cost/benefit" ratio. This, in the form "cost/confidence" ratio, is used by Garvey in his planning vision system. This measure is examined in Section 13.2.5; roughly, his "cost" was the effort in machine cycles to achieve goals, and his "confidence" approximated the probability of a goal achieving the correct outcome. The utility of correct outcomes was not explicitly encoded in his planner.

Sequential plan elaboration or partial plan elaboration can be interleaved with execution and scoring. Most practical planning is done in interaction with the world, and the plan scoring approach lends itself well to assessing such interactions. In Section 13.2.5 considers a planning vision system that uses enhanced plans and a limited replanning capability.

A thorny problem for decision making is to assess the cost of planning itself. The planning process is given its own utility (cost), and is carried only out as far as is indicated. Of course, the problem is in general infinitely recursive, since there is also the cost of assessing the cost of planning, etc. If, however, there is a known upper bound on the utility of the best achievable plan, then it is known that infinite planning could not improve it. This sort of reasoning is weaker than that needed to give the expected benefits of planning; it measures only the cost and maximum value of planning.

Another more advanced consideration is that the results of actions can be continuous and multidimensional, and discrete probabilities can be extended to probability distribution functions. Such techniques can reflect the precision of measurements.

An obviously desirable extension to a planner is a "learner," that can abstract rules for action applicability and remember successful plans. One approach would be to derive and remember ranges of planning parameters arising during execution; a range could be associated with a rule specifying appropriate action. This problem is difficult and the subject of current research.

13.2.4 Practical Simplifications

The expected utility calculations allow plans to be evaluated in a more or less "realistic" manner. However, in order to complete the calculations certain probabilities are necessary, and many of these reflect detailed knowledge about the interaction of phenomena in the world. It is thus often impractical to go about a full-blown treatment of scoring in the style of Section 13.2.2. This section presents some possible simplifications.

Of course, in many planning problems, such as those whose costs are nil or irrelevant, or all of whose goals are equally valuable, there is no need to address utility of plans at all. Such plans are typically not concerned with expenditure of real-world or planning resources.

Independence of various probabilities is one of the most helpful and pervasive assumptions in the calculation of probabilities. An example appeared in Section 13.2.2 with the table and telephone shape detectors.

Certain information can be ignored. Garvey [Garvey 1976] ignores failure information. His planning parameters include the "cost" of an action (strictly negative utilities reflecting effort), the probability of the action "succeeding," and the conditional probability that the state of the world is correctly indicated, given success. Related to ignoring some information is the assumption that certain outcomes are more reliable than others. For instance, the decision not to plan past "failure" reports means that they are assumed reliable.

Non-Bayesian rules of inference abound in planners [Shortliffe 1976]; the idea of assigning a single numerical utility score to plans is by no means the only way to make decisions.

13.2.5 A Vision System Based on Planning

Overview

This section outlines some features of a working vision system whose actions are controlled by the planning paradigm [Garvey 1976]. As with all large vision systems, more issues are addressed in this work than with the planning paradigm as a control mechanism. For one thing, the system uses multisensory input, including range and color information. An interactive facility aids in developing and testing low-level operators and "strategies" for object location. The machine-usable representation of knowledge about the objects in the scene domains and how they could be located is of course a central component.

The domain is office scenes (Fig. 13.6). For the task of locating different objects in such scenes, a "uniform strategy" is adopted. That is, the vision task is always broken down into a sequence of major goals to be performed in order. Such uniform strategies, if they are imposed on a system at all, tend to vary with different tasks, with different sensors or domain, or with different research goals.

Garvey's uniform strategy consists of the following steps.

1. *Acquire* some pixels thought to be in the desired region (the area of scene making up the image of the desired object).

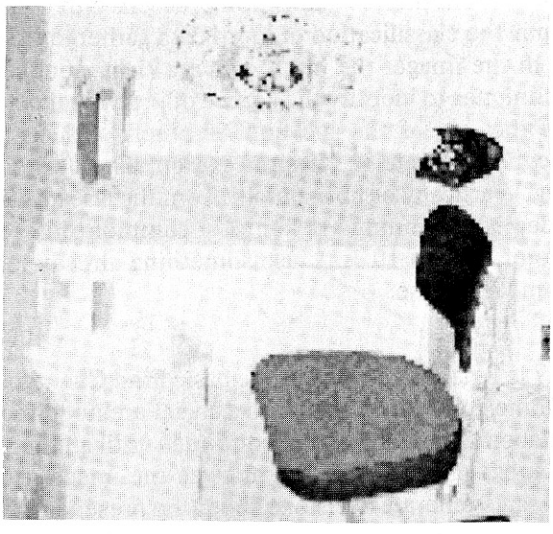


Fig. 13.6 The planning vision system uses input scenes such as these, imaged in different wavelengths and with a rangefinder.

2. *Verify* to some confidence that indeed the region was the desired one.
3. *Bound* the region accurately.

The outline the plan generation, scoring, and execution used in the system are described in the following paragraphs. The plans generated by the system are typically enhanced versions of plans like the telephone finder. Plan scoring proceeds as expected for such plans; allowances are made for the enhanced semantics of plan nodes. A "cost/confidence" scoring function is used, and various practical simplifications are made that do not affect the planning paradigm itself.

An Example Plan and Its Execution

The system's plans are enhanced plans, in the sense of Section 13.2.3. Actions can be *AND*, *OR* or *SEQUENCE* actions, and shared plan structure and loops are permitted. Loops that contain only internal, planning actions would never terminate. However, a loop with an OR node can terminate (has an exit) if one of the subactions of the OR is executable. A plan for locating a chair in an office scene is shown in Fig. 13.7. In Fig. 13.7, the acquire–validate–bound strategy is evident in the two SEQUENCE subgoals of the Find Chair main goal, which is an AND goal. The loop in the plan is evident, and makes sense here because often planning is done for information gathering, not for real world actions.

As noted in Section 13.2.3, an enhanced plan may not be completely specified. If it is to be executed one subgoal at a time (no parallelism is allowed), sequences of subactions must be determined for its AND and OR actions. In Garvey's planner, these sequences are determined initially on the basis of apriori information, but the partial results of actions are "fed back," so that dynamic rescoring and hence dynamic reordering of goal sequences is possible. For example, if one subgoal of an AND action fails, the AND action is abandoned. Thus this planner is to some degree incremental.

In execution, Fig. 13.7 might result in the sequence of actions depicted in Fig. 13.8. The acquisition phase of object location has the most alternatives, so plan generation effort is mainly spent there. Acquisition proceeds either directly or indirectly. Direct acquisition is the classification of input data gathered from a random sampling of a window in the image; the input data are rich enough to allow basic pattern recognition techniques to identify the source of individual pixels.

Indirect acquisition is the use of the location of other "objects" (really identified regions) in the scene to locate the desired region. The desired region might be found by "scanning" vertically or horizontally from the already identified region, for instance. The idea is a planning version of a common one (e.g., the geometric location networks of Section 10.3.2): use something already located to limit and direct search for something else.

Plan Generation

A plan such as Fig. 13.7 is "elaborated" from the basic Find Chair goal by recursively expanding goals. Some goals (such as to find a chair) are not directly executable; they need further elaboration. Elaboration continues until all the subgoals are executable. Executable subgoals are those that analyze the image, run filters and detectors over parts of it, and generate decisions about the presence or absence

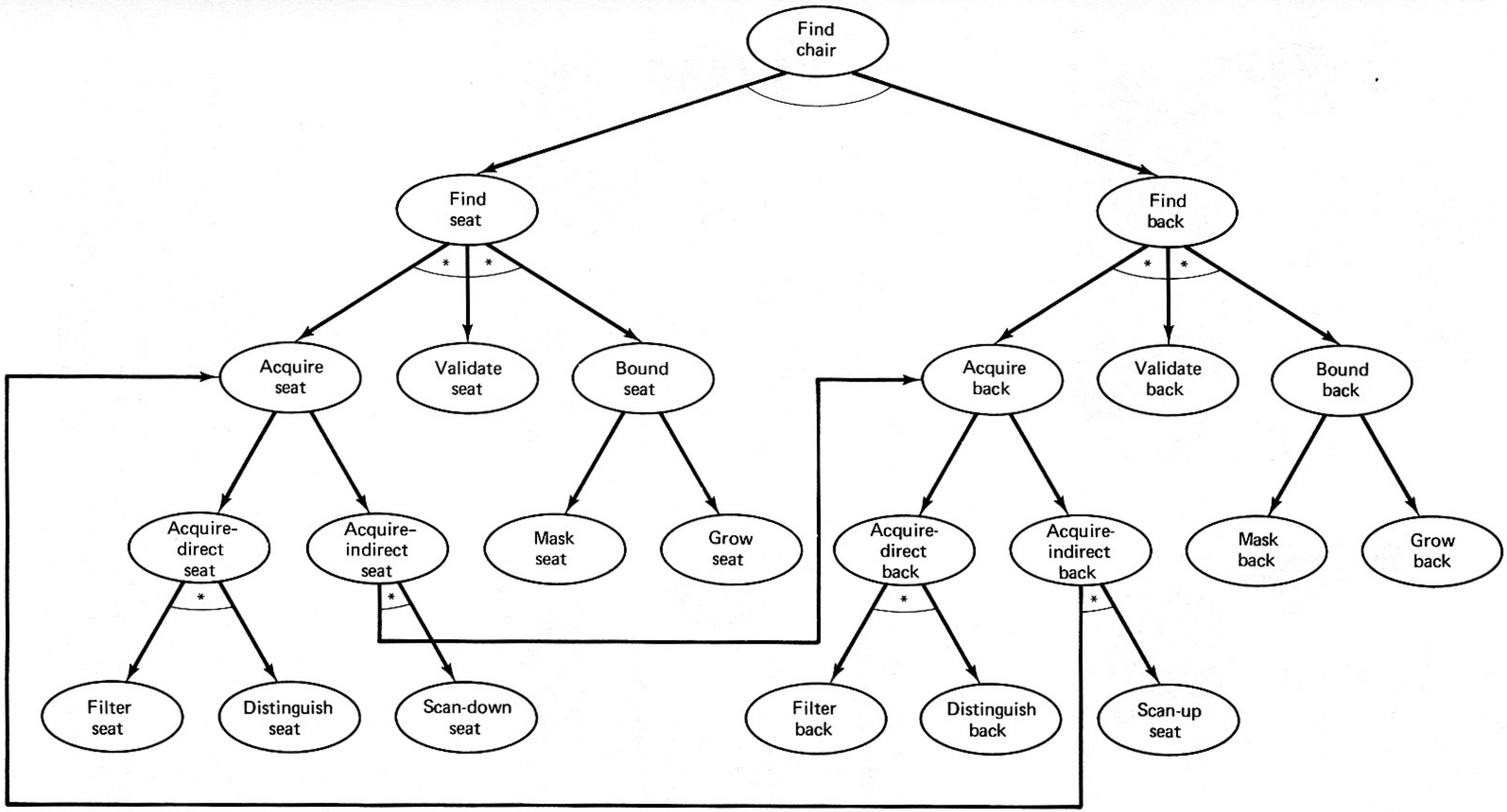
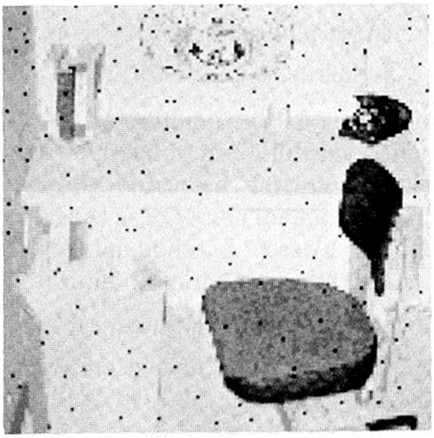
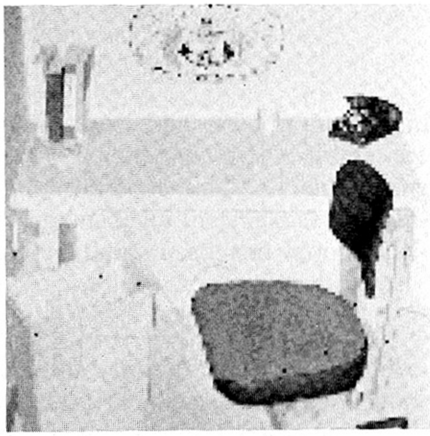


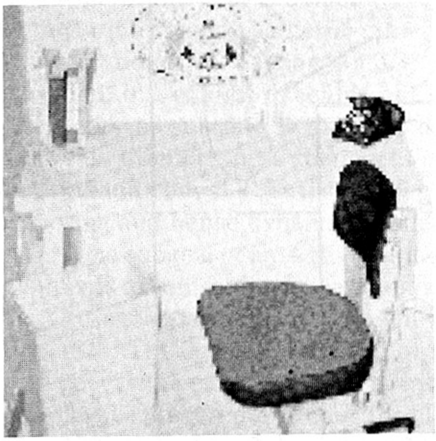
Fig. 13.7 An enhanced plan to locate a chair in an office scene. Untied multiple arcs denote OR actions, arcs tied together denote AND actions, those with *'s denote SEQUENCE actions. The loop in the plan has executable exits.



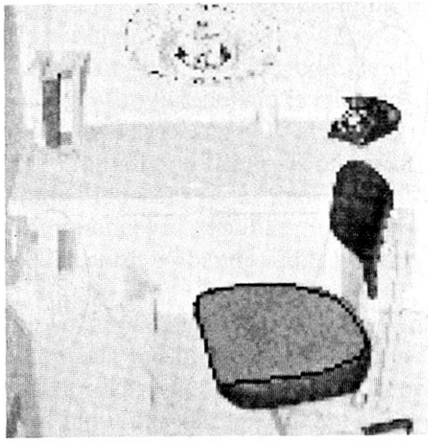
(a)



(b)



(c)



(d)

Fig. 13.8 The plan of Fig. 13.7 finds the most promising execution sequence for finding the chair in the scene of Fig. 13.6: find the seat first, then scan upwards from the seat looking for the back. Acquisition of the seat proceeds by sampling (a), followed by classification (b). The Validation procedure eliminates non-chair points (c), and the Bounding procedure produces the seat region (d). To find the back, scanning proceeds in the manner indicated by (e) (actually fewer points are examined in each scan). The back is acquired and bounded, leading to the final location of the chair regions (f).

(e) (f)

Fig. 13.8 (cont.)

of image phenomena. This straightforward elaboration is akin to macro expansion, and is not a very sophisticated planning mechanism (the program cannot criticize and manipulate the plan, only score it). A fully elaborated plan is presented for scoring and execution.

The elaboration process, or planner, has at its disposal several sorts of knowledge embodied as modules that can generate subgoals for a goal. Some are general (to find something, find all its parts); some are less general (a chair has a back and a seat); some are quite specific, being perhaps programs arising from an earlier interactive method-generation phase. The elaborator is guided by information stored about objects, for instance this about a tabletop:

OBJECT	PROPERTIES	RELATIONS
Table TOP	Hue:26–58	Supports Telephone 0.6
	Sat.: 0.23–0.32	Supports Book 0.4
	Bright.: 18–26	Occludes Wall 1
	Height: 26–28	
	Orient.: −7–7	

Here the orientation information indicates a vertical surface normal. The planner knows that it has a method of locating horizontal surfaces, and the plan elaborator can thus create a goal of direct acquisition by first locating a horizontal plane. The relational information allows for indirect acquisition plans. The elaborator puts direct and indirect alternatives under an *OR* node in the plan. Information not used for acquisition (height, color) may be used for validation.

Loops may occur in an elaborated plan because each newly generated goal is checked against goals already existing. Should it or an equivalent goal already exist, the existing goal is substituted for the newly generated one. Goals may thus have more than one ancestor, and may depend on one another.

At this stage, the planner does not use any planning parameters (cost, utilities, etc.); it is strictly symbolic. As mentioned above, important information about execution sequences in an enhanced plan is provided by scoring.

Plan Scoring and Execution

The scoring in the vision plan is a version of that explained in Sections 13.2.2 through 13.2.4. Each action in a plan is assumed either to succeed (S) in locating an object or to fail. Each action may report either success ("S") or failure. An action is assumed to report failure correctly, but possibly to be in error in reporting success. Each action has three "planning parameters" associated with it. They are C, its "cost" (in machine cycles), $P(\text{"}S\text{"})$ the probability of it reporting success, and $P(S|\text{"}S\text{"})$, the probability of success given a report of success.

As shown earlier, the product

$$P(S|\text{"}S\text{"})P(\text{"}S\text{"}) \tag{13.19}$$

is the probability that the action has correctly located an object and reported success. This product is called the "confidence" of the action. An action has structure as shown in Fig. 13.9.

The score of an action is computed as

$$\text{score} = \frac{\text{cost}}{\text{confidence}} \tag{13.20}$$

The planner thus must minimize the score.

The initial planning parameters of an executable action typically are determined by experimentation. The parameters of internal (AND, OR, SEQUENCE) actions by scoring methods alluded to in Sections 13.2.2, 13.2.3, and the Exercises (there are a few idiosyncratic ad hoc adjustments.).

It may bear repeating that planning, scoring, and execution are not separated temporally in this sytem. Scoring is used after the enhanced plan is generated to derive a simple plan (with ordered subgoals). Execution can affect the scores of nodes, and so execution can alternate with "replanning" (really rescoring resulting in a reordering). Recall the example of failure of an *AND* or *SEQUENCE* subgoal, which can immediately fail the entire goal. More generally, the entire goal and ultimately the plan may be rescored. For instance, the parameters of a successful action are modified by setting the cost of the executed action to 0 and its confidence to its second parameter, $P(S|\text{"}S\text{"})$.

Given a scored plan, execution is then easy; the execution program starts at the top goal of the plan, working its way down the best path as defined by the scores of nodes it encounters. When an executable subgoal is found (e.g. "look for a green region"), it is passed to an evaluation function that "runs" the action associated with the subgoal.

The subgoal is either achieved or not; in either case, information about its outcome is propagated back up the plan. Failure is easy; a failed subgoal of an AND or SEQUENCE goal fails the goal, and this failure is propagated. A failed subgoal of an OR goal is removed from the plan. The use of success information is more complex, involving the adjustment of confidences and planning parameters illustrated above.

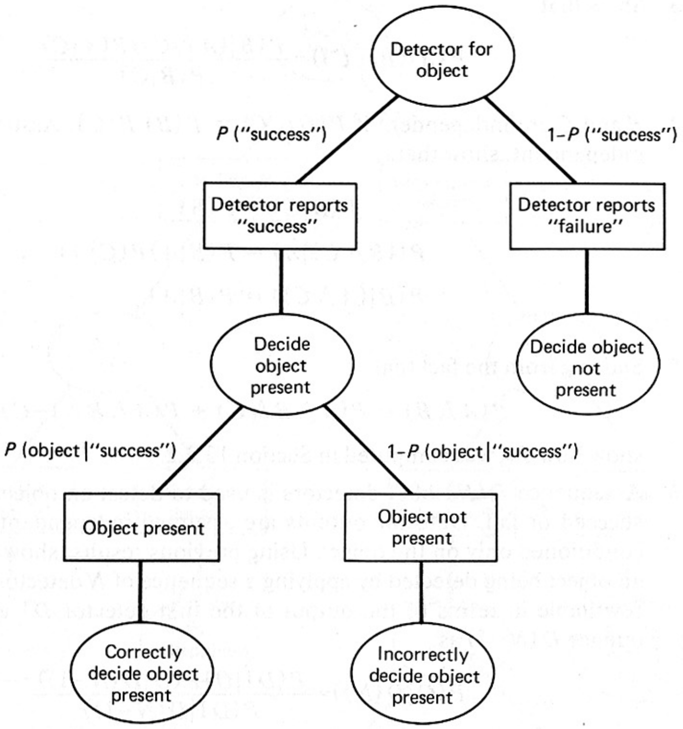


Fig. 13.9 This is the microstructure of a node ("action") of Garvey's planning system in terms of simple plans. Think of actions as being object detectors which announce "Found" or "Not Found." Garvey's planning parameters are P("Found") and P(Object is there|"Found"). Confidence in the action is their product; it is the probability of correctly detecting the object. All other outcomes are lumped together and not used for planning.

After the outcome of a goal is used to adjust the parameters of other goals, the plan is rescored and another cycle of execution performed. The execution can use knowledge about the image picked up along the way by prior execution. This is how results (such as acquired pixels) are passed to later processing stages (such as the validation process). Such a mechanism can even be used to remember successful subplans for later use.

EXERCISES

13.1 Complete the computation of outcome probabilities in the style of Section 13.2.2, using the assumptions given there. Check your work by showing (symbolically) that the probabilities of getting to the terminal actions ("goal states") of the plan sum to 1.

13.2 Assume in Section 13.2.2 that the results of the "table" and "telephone shape" detectors are not independent. Formulate your assumptions and compute the new outcome probabilities for Fig. 13.4.

13.3 Show that

$$P(A \mid (B \wedge C)) = \frac{P(B \mid (A \wedge C))P(A \mid C)}{P(B \mid C)}$$

13.4 B and C are independent if $P(B \wedge C) = P(B)\,P(C)$. Assuming that B and C are independent, show that

$$P(B \mid C) = P(B)$$
$$P((B \wedge C) \mid A) = P(B \mid A)P(C \mid A)$$
$$P(B \mid (A \wedge C)) = P(B \mid A)$$

13.5 Starting from the fact that

$$P(A \wedge B) = P(A \wedge B \wedge C) + P(A \wedge B \wedge (-C))$$

show how P_{15} was computed in Section 13.2.2.

13.6 A sequence $D(N)$ of N detectors is used to detect an object; the detectors either succeed or fail. Detector outputs are assumed independent of each other, being conditioned only on the object. Using previous results, show that the probability of an object being detected by applying a sequence of N detectors $D(N)$ is recursively rewritable in terms of the output of the first detector $D1$ and the remaining sequence $D(N-1)$ as

$$P(O \mid D(N)) = \frac{P(D1 \mid O)P(O \mid D(N-1))}{P(D1 \mid D(N-1))}$$

13.7 Consider scoring a plan containing an OR node (action). Presumably, each subgoal of the OR has an expected utility. The OR action is achieved as soon as one of the subgoals is achieved. Is it possible to order the subgoals for trial so as to maximize the expected utility of the plan? (This amounts to a unique "best" rewriting of the plan to make it a simple plan.)

13.8 Answer question 13.7 for an AND node; remember that the AND will fail as soon as any of its subgoals fails.

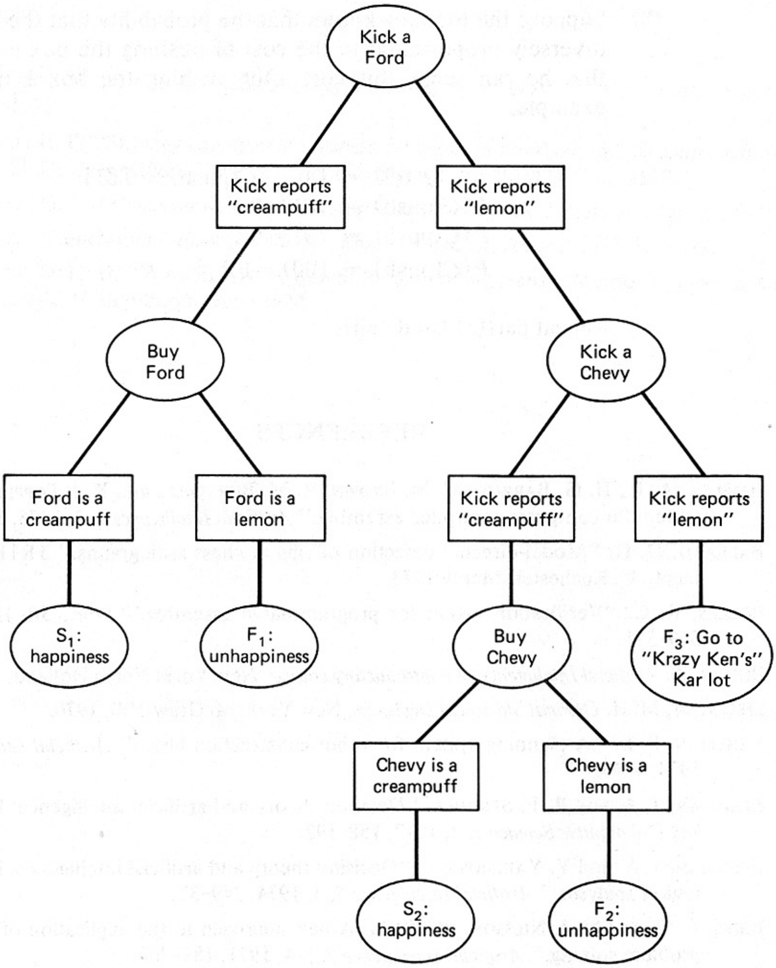
13.9 What can you say about how the cost/confidence ratio of Garvey's planner is related to the expected utility calculations of Section 13.2.2?

13.10 You are at Dandy Dan's used car lot. *Consumer Reports* says that the a priori probability that any car at Dandy Dan's is a lemon is high. You know, though, that to test a car you kick its tire. In fact, with probability:

$P(\text{"}C\text{"} \mid C)$	: a kick correctly announces "creampuff" when the car actually is a creampuff
$P(\text{"}C\text{"} \mid L)$	: a kick incorrectly announces "creampuff" when the car is actually a lemon
$P(L)$	: the a priori probability that the car is a lemon

Your plan for dealing with Dandy Dan is shown below; give expressions for the probabilities of arriving at the nodes labeled S_1, S_2, F_1, F_2, and F_3. Give numeric answers using the following values

$$P(\text{"}C\text{"} \mid C) = 0.5,\ P(\text{"}C\text{"} \mid L) = 0.5,\ P(L) = 0.75$$



Ex. 13.10

13.11 Two bunches of bananas are in a room with a monkey and a box. One of the bunches is lying on the floor, the other is hanging from the ceiling. One of the bunches is made of wax. The box may be made of flimsy cardboard. Given that:

$P(\mathrm{WH}) = 0.2$: probability that the hanging bananas are wax
$P(\mathrm{WL}) = 0.8$: probability that the lying bananas are wax
$P(\mathrm{C}) = 0.5$: probability that the box is cardboard
$U(\mathrm{eat}) = 200$: utility of eating a bunch of bananas
$C(\mathrm{walk}) = -10$: cost of walking a unit distance
$C(\mathrm{push}) = -20$: cost of pushing the box a unit distance
$C(\mathrm{climb}) = -20$: cost of climbing up on box

(a) Analyze two different plans for the monkey, showing all paths and calculations. Give criteria (based upon extra information not given here) that would allow the monkey to choose between these plans.

(b) Suppose the monkey knows that the probability that the box will collapse is inversely proportional to the cost of pushing the box a unit distance (and that he can sense this cost after pushing the box 1 unit distance). For example,

$$
\begin{aligned}
P(C) &= 1.0 - [C(\text{push}) \times 0.01] \\
P(C(\text{push}) &= 10) = 0.1 \\
P(C(\text{push}) &= 20) = 0.1 \\
P(C(\text{push}) &= 100) = 0.1
\end{aligned}
$$

Repeat part(a) (in detail).

REFERENCES

AMBLER, A. P., H. G. BARROW, C. M. BROWN, R. M. BURSTALL, and R. J. POPPLESTONE. "A versatile system for computer controlled assembly." *Artificial Intelligence 6*, 2, 1975, 129–156.

BALLARD, D. H. "Model-directed detection of ribs in chest radiographs." TR11, Computer Science Dept., U. Rochester, March 1978.

BOLLES, R. C. "Verification vision for programmable assembly." *Proc.*, 5th IJCAI, August 1977, 569–575.

BUNDY, A. *Artificial Intelligence: An introductory course.* New York: North Holland, 1978.

DEGROOT, M. H. *Optimal Statistical Decisions.* New York: McGraw-Hill, 1970.

FAHLMAN, S. E. "A planning system for robot construction tasks." *Artificial Intelligence 5*, 1, Spring 1974, 1–49.

FELDMAN, J. A. and R. F. SPROULL. "Decision theory and artificial intelligence: II. The hungry monkey." *Cognitive Science 1*, 2, 1977, 158–192.

FELDMAN, J. A. and Y. YAKIMOVSKY. "Decision theory and artificial intelligence: I. A semantics-based region analyser." *Artificial Intelligence 5*, 4, 1974, 349–371.

FIKES, R. E. and N. J. NILSSON. "STRIPS: A new approach to the application of theorem proving to problem solving." *Artificial Intelligence 2*, 3/4, 1971, 189–208.

FIKES, R. E., P. E. HART, and N. J. NILSSON. "New Directions in robot problem solving." In *M17*, 1972a.

FIKES, R. E., P. E. HART, and N. J. NILSSON. "Learning and executing generalized robot plans." *Artificial Intelligence 3*, 4, 1972b, 251–288.

GARVEY, J. D. "Perceptual strategies for purposive vision." Technical Note 117, AI Center, SRI Int'l, 1976.

MACKWORTH, A. K. "Vision research strategy: Black magic, metaphors, mechanisms, mini-worlds, and maps." In *CVS*, 1978.

MCCARTHY, J. and P. J. HAYES. "Some philosophical problems from the standpoint of artificial intelligence." In *M14*, 1969.

NILSSON, N. J. *Principles of Artificial Intelligence.* Palo Alto, CA: Tioga Publishing Company, 1980.

RAIFFA, H. *Decision Analysis.* Reading, MA: Addison-Wesley, 1968.

SACERDOTI, E. D. "Planning in a hierarchy of abstraction spaces." *Artificial Intelligence 5*, 2, 1974, 115–135.

SACERDOTI, E. D. *A Structure for Plans and Behavior.* New York: Elsevier, 1977.

SHIRAI, Y. "Analyzing intensity arrays using knowledge about scenes." In *PCV*, 1975.

SHORTLIFFE, E. H. *Computer-Based Medical Consultations: MYCIN.* New York: American Elsevier, 1976.

SPROULL, R. F. "Strategy construction using a synthesis of heuristic and decision-theoretic methods." Ph.D. dissertation, Dept. of Computer Science, Stanford U., May 1977.

SUSSMAN, G. J. *A Computer Model of Skill Acquisition.* New York: American Elsevier, 1975.

TATE, A. "Generating project networks." *Proc.*, 5th IJCAI, August 1977, 888–983.

WARREN, D. H. D. "WARPLAN: A system for generating plans." Memo 76, Dept. of Computational Logic, U. Edinburgh, June 1974.