COGNITIVE PSYCHOLOGY 17, 26–65 (1985)

Discrimination of Operator Schemata in Problem Solving: Learning from Examples

MATTHEW W. LEWIS AND JOHN R. ANDERSON

Carnegie-Mellon University

Three studies investigate how problem solvers learn to apply appropriate actions in problem solving. Part of this knowledge appears to result from learning the sets of problem features (schemata) that predict the success of different problem-solving actions (operators). A major claim is that this learning can be produced, in part, by the same mechanisms that produce concept formation and abstraction of object schemata. Studies using geometry proof problems and an abstract maze-searching task produce results similar to common findings in the schema abstraction literature: Performance improves as the prototypicality of the feature sets increases. Also examined are the effects of active processing of problem features, relevance of the discriminating features to the problem solution, amount of practice, and delay of feedback regarding the accuracy of operator selection. Subjects learn when to apply an operator better during active, deliberate hypothesis testing, regardless of feature relevance. Delayed feedback produces poorer performance with extended practice. A model of classification learning and a simulation of explicit hypothesis testing produced reasonable fits to the data. The failure to find evidence for unconscious learning is evidence against the automatic discrimination mechanism proposed in ACT*.

How is problem-solving expertise acquired by solving example problems? Somehow problem solvers in complex domains learn to attend to the important features of a problem and then select the action which will lead to a solution of that problem. This phenomenon occurs in all disciplines—from chess grand masters choosing a move (Chase & Simon, 1973) through successful students of geometry selecting a strategy for solving a proof problem (Anderson, Greeno, Kline, & Neves, 1980; Greeno, 1978) to expert VW mechanics diagnosing and executing an engine repair. As educators we would like to make the acquisition of expertise more efficient. As psychologists our interests lie in defining the processes which underlie complex skill acquisition.

This research was supported by Grant 1ST-80-15357 from the Information Sciences Program of the National Science Foundation. We thank Frank Boyle, Gary Bradshaw, Seth Chaiklin, Renée Elio, Rob Farrell, Bill Jones, Jill Larkin, Peter Pirolli, and Miriam Schustack for useful input throughout the project, Marilyn Jacobs for editing assistance, and Jeff Shrager for his computer wizardry. Special thanks go to Catherine Jatkowski of Schenley High School, Pittsburgh, PA, for her valuable help and advice. Send requests for reprints to Matthew Lewis, Learning Research and Development Center, University of Pittsburgh, Pittsburgh, PA 15213.

0010-0285/85 $7.50

An important component of skill acquisition is refining, or tuning, the conditions under which an operator should apply (Anderson, 1981). An operator is defined as a problem-solving action such as the use of the side–angle–side postulate in geometry. Sometimes applying such an operator will take the problem solver one step closer to a solution and sometimes it will not. On the basis of three studies we argue that subjects acquire schemata for where operators will and will not work. These schemata are abstracted from example problems. Such a schema consists of features of the problem that are predictive of the success of the operator.

A major claim of this paper is that the process of forming these operator schemata has much in common with concept formation and abstraction of object schemata. In both concept formation and schema abstraction one acquires the ability to apply a label to examples of categories. Concept formation occurs with well-defined categories (Bower & Trabasso, 1964; Bruner, Goodnow, & Austin, 1956; Hunt, Marin, & Stone, 1966; Levine, 1966). Schema abstraction refers to the acquisition of concepts that are not well defined, i.e., not governed by a single rule (Anderson, 1980; Anderson, Kline, & Beasley, 1979; Elio & Anderson, 1981; Hayes-Roth & Hayes-Roth, 1977; Mervis & Pani, 1980; Posner, 1970; Reed, 1972; Rosch, 1978). From the subject's view, the relationship between the cues and the categories may be probabilistic even if they are deterministic from the experimenter's view. The concept-learning literature has shown that probabilistic relationships between categories and cues leads to very dramatic reductions in learning rates. Similarly, problem solvers have trouble learning when to apply operators in domains like chess and geometry. Our hypothesis is that acquisition of operator schemata involves basically the same processes as are involved in acquisition of object schemata. To support this position we present evidence of phenomena found in the learning of problem-solving operators that are qualitatively and quantitatively analogous to phenomena found in abstraction of object schemata.

OPERATOR SCHEMATA IN GEOMETRY

Proof generation problems in geometry contain certain problem features which are correlated with useful operators. These features are properties of the diagram and information in the problem statement. The operators in this case are rules of inference to be used in the proof. After a brief review of high school geometry proof problems we present evidence of a correlation between surface features of geometry problems and the correct rules of inference for these problems.

A common proof problem in high school geometry involves proving that two triangles (e.g., $\triangle ABC$ and $\triangle DEF$) are congruent ($\cong$) to each other. The problem statement in such a proof generally contains a diagram, a list of information (a set of hypotheses or "givens") which is given as true, and a statement to be proven (the conclusion). The general goal is to use deductive reasoning to show that the statement to be proven follows from the hypotheses given and a set of inference rules. In the case of proving two triangles to be congruent this is achieved by showing that corresponding parts of the two triangles are congruent to each other. Triangles have two types of components which can be shown to be congruent; sides and angles. Sides are line segments with endpoints. The line segment from point A to B is written as $\overline{AB}$. Angles in triangles are line segments which are noncollinear and share a common endpoint. The angle with point A as its vertex is written as $\angle BAC$ or $\angle CAB$.

By using postulates of geometry one can infer that two triangles are congruent after only a subset of their sides and angles have been shown to be congruent. An example of this set of postulates is the side–angle–side (SAS) postulate; if it can be shown that two sides and the included angle are congruent to the corresponding sides and angle of another triangle, then one can infer that the triangles are congruent.

In some simple geometry problems enough information is given explicitly in the problem statement to identify the correct operator. For instance, if the student is asked to prove $\triangle ABC \cong \triangle DEF$ and is given $\overline{AB} \cong \overline{DE}$, $\overline{BC} \cong \overline{EF}$, and $\overline{CA} \cong \overline{FD}$, then these givens identify the side–side–side (SSS) postulate for proving triangles congruent as the appropriate operator. That is, the three features are necessary and sufficient for the rule to be applied. This is a particularly clear case of a schema; however, few problems are so simple. Usually, the proof involves several steps of inference rule application and there are no simple sets of features that identify which rules are needed for the proof. The solver does not know if the prior choice of inference rules is correct until the problem is completely solved. In those cases there may still be a correlation between features and operators, just as there is a correlation between flying and being a bird: The feature of flying is only probabilistic in terms of its diagnosticity for something being a bird. Some things that fly are not birds and some birds do not fly. Similarly, for a surface feature that signals SAS, other inference rules will sometimes apply appropriately in the presence of that feature. The point is that the structure or semantics of the geometry domain creates correlations between features and operators. The learner can assemble the features which predict correct use of an operator into a schema for that operator.

We examined a locally used high school geometry textbook by Jurgensen, Donnelly, Maier, and Rising (1975) for evidence of feature–operator correlations. One type of problem examined required students to

Given: $\angle 1 \cong \angle 2$
$\overline{AC} \cong \overline{BD}$
$\overline{BK} \cong \overline{CK}$

Prove: $\triangle ABK \cong \triangle DCK$

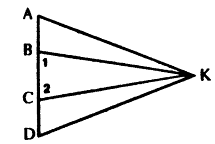

FIG. 1. Sample geometry proof problem with correlation of problem features to proof strategy, as recommended by the teacher's manual. Adapted, by permission of the publisher, from Jurgensen et al. (1975).

make an inference about triangle congruence. Problems where the appropriate inference rules were either SSS, SAS, or ASA (angle–side–angle) were checked for surface features that predicted the correct (recommended by the teacher's manual) inference rule.

Consider Fig. 1 as an example problem. This is most directly solved by using the SAS postulate. Note that the givens involve two sides and an angle, although only one of these sides is actually part of the SAS configuration to be proven. Is there any connection between whether sides and angles are given as congruent and the inference method chosen? Table 1 shows an analysis of the correlation between angle or side congruences given in such problems and the solution strategy recommended by the teacher's manual. As in Fig. 1, these congruences are not always of sides and angles which are part of the to-be-proven congruent triangles. Problem frequencies are recorded in each cell of the table. There is clearly a correlation between features of the givens and solution strategy. A student could simply count angle and side congruences and have a fair idea of how to solve the problem.

TABLE 1
Categorization of Triangle Congruence Proof Problems from Jurgensen et al. (1975) Showing Correlation of Features in the Given Statements with Solution Strategy

Correct solution strategy	Congruences in the given statements					
	AA	AS	SS	A	S	0
ASA	17	7	0	6	7	3
SAS	0	13	6	1	6	3
SSS	0	0	12	0	2	0

Note. Numbers indicate the number of problems in each cell. AA refers to all problems with at least two pairs of angles given as congruent; AS refers to all problems with just one angle congruence and at least one side congruence; SS refers to all problems with no angle congruences and at least two side congruences; A refers to problems with a single angle congruence and no side congruences; S refers to problems with a single side congruence and no angle congruences; 0 refers to all problems with no side and no angle congruences given.

Given: $\overline{AB}$ bisects $\overline{CD}$
$\overline{CD}$ bisects $\overline{AB}$
$\overline{AB} \perp \overline{CD}$

Prove: $\triangle CAX \cong \triangle DBX$

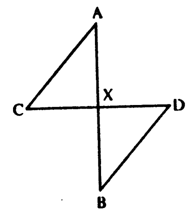

FIG. 2. Sample geometry proof problem with correlation of problem features to whether vertical angles should be used in the proof, as recommended by the teacher's manual. Adapted, by permission of the publisher, from Jurgensen et al. (1975).

Table 2 contains counts of geometry problems containing a vertical angle configuration (two angles whose sides form two pairs of collinear line segments or opposite rays). The vertical angles in the problem are or are not used in the final proof. Figure 2 illustrates one such problem. The teacher's manual recommends that this problem be solved without inferring that the vertical angles are congruent. What features predict whether it will be useful to make the vertical angle inference? There is a strong positive correlation between whether the vertical angles are both part of the to-be-proven triangles and are also not formed by perpendicular lines, and making the inference that the vertical angles are congruent. There is a corresponding negative correlation with the angles not being part of the to-be-proven triangles and not being formed by perpendicular lines. Again, surface features exist which could be used to guide the inferencing which is necessary to solve the proof.

While these correlations between problem features and operators are

TABLE 2
Categorization of Triangle Congruence Proof Problems from Jurgensen et al. (1975) Showing Correlation of Need to Infer that Vertical Angles Are Congruent with Other Features of the Problem

	Necessary to make the inference that two vertical angles are congruent in a triangle congruence problem?	
	Yes	No
Angles are		
Part of to-be-proven triangles and are perpendicular	0	6
Part of to-be-proven triangles and are not perpendicular	8	4
Not part of to-be-proven triangles and are perpendicular	0	2
Not part of to-be-proven triangles and are not perpendicular	1	12

not necessary, they are certainly reasonable. It would be strange if the SSS postulate was used more often when no information about segment congruence was given. It would also be strange if vertical angles were only used when they were not part of to-be-proven congruent triangles. It is an open question whether subjects' problem solving is actually guided by such surface correlations. With practice, subjects appear to become more accurate in selecting inference rules, but this may reflect some deeper logical analysis of the problem structure. One purpose of this research is to determine if and when problem solving can be controlled by surface features that do not have any deeper justification.

Do problem solvers use surface features to guide operator selection? There is evidence in the literature which suggests the problem solvers in algebra and physics use surface features of problems to direct problem solving. The existence of categories or schemata for algebra word problems has been argued by Mayer, both on the basis of a recent survey of textbook word problems (Mayer, 1981) and from results of a recall task (Mayer, 1982). Silver (1981) found that high-ability seventh-grade students tend to recall information about the mathematical structure of word problems they have solved, whereas low-ability students tend to recall information about the cover story or context of the problem. When categorizing word problems the same pattern of performance appears: High-ability students sort problems on the basis of common problem structure, whereas poor problem solvers tended to form groups based on the common surface details of the problems. Studies of algebra word problem solving by Hinsley, Hayes, and Simon (1977) provide evidence that problem solvers categorize algebra word problems early in reading the problem, sometimes based on as little as the initial noun phrase of the problem. Based on the category, the subjects then choose a representation and processing strategy for the problem. Studies of expert–novice differences in physics problem solving (Chi, Feltovich, & Glaser, 1981; Larkin, McDermott, Simon, & Simon, 1980) have found that experts use abstract physics principles to approach and represent a problem whereas novices tend to base their representations and approaches on the literal features of problems.

Some problem solvers do, then, appear to use surface features to guide problem solving. The phenomenon appears with particular salience in the performance of novices because they often do not attend to the correct features: This leads to observable errors.[1] However, in some cases, ex-

[1] The efficacy of attending only to surface features may be due to the simplicity of the problems encountered by novices early in their learning. The correlation of surface features to an operator may be more salient than the correlation of abstract structural features to an operator.

perts may also use surface features to select problem-solving operators without accessing the underlying reasons for *why* an operator is appropriate. Experts evidently have discriminated when it is safe to rely only on surface features without accessing the deeper structure of the problem. Their feature-to-operator correlations have changed with experience. For example, most proficient simplifiers of algebra equations know what to do in order to simplify the following equation:

$$\frac{3}{X} = \frac{X}{4}.$$

The most efficient operator to apply is cross multiplication. When asked why that is the correct operator, many people reasonably reply that they want to get rid of the fractions, so cross multiplying is "what you do." They can state the relevant goal and operator but can seldom spontaneously explain why cross multiplying works in this situation. The people who are able to explain usually generate a post hoc explanation based on their knowledge of mathematics. Perhaps at one point they were taught why this was a legal operation but it may have been taught only as an algorithm. The point here is that experts appear to use problem features to guide problem solving without accessing the underlying structure of the problem domain.

While the evidence is clear that surface features can guide problem solving, the evidence is unclear about how surface features acquire this power. Is it the result of learners encoding feature–operator correlations or does it depend on some analysis of the problem structure? This question is addressed in Experiment 1 where subjects solved a set of geometry proof problems which contained arbitrary correlations between surface features and inference rules. Subjects were then tested for the degree to which these correlations were correctly learned. In posing this question we do not intend to imply that if subjects use arbitrary surface features they may not also take advantage of meaningful connections between features and operators, when such features exist.

Experiments 2 and 3 explore two questions regarding the connections between features and operators. First, to what degree are operator schemata similar to object schemata? The second question addresses the effects of feedback delay on learning operator schemata. This question arises from an interest in exploring the conditions that are favorable to schema abstraction for operators.

THE ACT* THEORY OF OPERATOR DISCRIMINATION

Our questions concerning operator abstraction, although independently interesting, were motivated by the ACT* theory (Anderson, 1982, 1983a)

of how such operators are acquired.[2] According to ACT*, a subject acquires various productions that correspond to the rules of a domain. Simply defined, a production is a condition–action rule which performs its action when its conditions are met. Thus, from the SAS postulate the subject could construct the following rule:

P1: IF the goal is to prove $\Delta XYZ \cong \Delta UVW$
THEN try the SAS postulate and set as subgoals to
1. Prove $\overline{XY} \cong \overline{UV}$
2. Prove $\angle XYZ \cong \angle UVW$
3. Prove $\overline{YZ} \cong \overline{VW}$

Such a rule will sometimes successfully solve the problem and sometimes fail. If a production fails to solve a problem, an attempt is made to add some simple features to the conditions of that production which will discriminate its success from its failure. As more conditions are added to a production its specificity or precision of application increases. For instance, if a side congruence was present when the rule applied and succeeded in solving a problem and was not present when the rule applied and failed to solve a problem, the following discrimination would be added to the production set:

P2: IF the goal is to prove $\Delta XYZ \cong \Delta UVW$
and a side congruence is contained in the givens
THEN try the SAS postulate and set as subgoals to
1. Prove $\overline{XY} \cong \overline{UV}$
2. Prove $\angle XYZ \cong \angle UVW$
3. Prove $\overline{YZ} \cong \overline{VW}$

This more specific rule might later misapply, and the discrimination process would reoccur. If unsuccessful application occurred again it might be noted that, in addition to the side congruence, an angle congruence was given. Discrimination would then produce the following production rule:

P3: IF the goal is to prove $\Delta XYZ \cong \Delta UVW$
and a side congruence is contained in the givens
and an angle congruence is given
THEN try the SAS postulate and set as subgoals to
1. Prove $\overline{XY} \cong \overline{UV}$
2. Prove $\angle XYZ \cong \angle UVW$
3. Prove $\overline{YZ} \cong \overline{VW}$

[2] For a review of discrimination learning in ACT*, see Anderson, 1982.

Note that these three rules can coexist but P3 would be used, if applicable, because it has the most specific condition that is fully satisfied.

In overview, then, the discrimination analysis implies that learners may know the general operators of the domain but may not know when to apply them. This is consistent with the general observation that both teachers and textbooks teach general rules, like the side–angle–side postulate, but provide no strategic advice about when to use them. In analogy to the concept-learning literature, this is like knowing that there exists a type A person with certain features and yet not being able to identify a type A person from among others.

In addition to testing the general position that acquisition of operator schemata involves the same processes underlying concept formation and abstraction of object schemata, the following experiments test three important qualitative features of ACT*'s theory of operator discrimination:

1. Discrimination in ACT* is automatic; it does not depend on a logical or conscious analysis of feature–operator correlations. ACT* predicts that subjects will learn the feature–operator correlations even if the features are only incidentally correlated to the operators. This is because discrimination is a basic learning mechanism in ACT* and such basic mechanisms are not subject to strategic control. These mechanisms are hypothetically "wired-in" operators that are part of the basic architecture.

2. The discrimination process in ACT* is invoked only when an operator is applied incorrectly. Discrimination is part of the general learning-by-doing philosophy of ACT* and is an attempt to correct a misfunctioning system. Discriminations will not occur if the subject passively observes feature–operator correlations. For discrimination of an operator to occur the subject must actively apply the operator and get feedback about the operator's success or failure. Only when the application fails will discrimination learning take place. In this, the ACT* theory of how feature schemata are formed differs from theories of observational learning of categories (Brooks, 1978; Reber, 1976).

3. Performance will improve as the stimulus becomes more prototypical. ACT* agrees with most schema abstraction theories in this prediction. In ACT*, the more features a stimulus contains that are correlated with an operator, the more likely that stimulus will be to evoke application of that operator. The presence of many correlated features will evoke highly specific and accurately learned operators like P3.

Experiment 1 tests predictions 1, 2, and 3 above. Experiments 2 and 3 look in detail at prediction 3.

EXPERIMENT 1

Subjects in Experiment 1 studied and solved 20 geometry problems, 10 of which could be solved by the SAS postulate and 10 of which could be solved by the ASA postulate. Four problem features in the problems were varied. Two of the features occurred 80% of the time with SAS and 20% of the time with ASA. The other two features occurred 80% of the time with ASA and 20% of the time with SAS. This was intended to replicate the partial correlations that occur in the geometry text we examined. However, these feature–operator correlations were arbitrary and, in fact, were reversed for half the subjects. After they studied and solved these problems, subjects were tested for transfer to briefly presented problems that could be solved by both or by neither strategy. In addition to testing whether subjects could acquire feature–operator correlations in the form of operator schemata, we attempted to vary two variables; the subjects' use of active hypothesis testing while problem solving and how relevant or semantically related the problem's predictive features are to the problem and solution.

Does the degree of active processing or conscious hypothesis testing affect how well schemata are acquired? The general effect of active processing on learning, by means of verbalizing the reasons for behavior in a problem-solving task, has been studied: The general result is improved immediate performance and learning (Davis, Carey, Foxman, & Tarr, 1968; Gagné & Smith, 1962; Wilder & Harvey, 1971). In a seminal study, Gagné and Smith (1962) found that instructing subjects to verbalize why they were making each move while learning to solve the Tower of Hanoi puzzle had significant effects. Subjects who were forced to verbalize while solving puzzles in a learning phase showed significant reductions in the number of moves in excess of the minimum number needed to solve a transfer puzzle, as compared to control condition subjects who solved the learning puzzles without verbalization. The subjects who verbalized also were significantly faster at solving the puzzle.

How does verbalization improve learning? Results similar to those above are explained by Ericsson and Simon (1980) as examples of verbalizations changing the course of problem solving. Subjects are forced to make inferences about their mental processes and hence access information which would not be accessed if these inferences were not made. It is the attention to this extra information which hypothetically accounts for the improved performance and learning. Perhaps part of that extra information accessed is simply more explicit and specific generation of hypotheses.

A related explanation for how verbalization could improve learning comes from work investigating how different problem-solving strategies affect the acquisition of information about problem structure (Sweller, 1983; Sweller & Levine, 1982; Sweller, Mawer, & Ward, 1983). The general findings of this work suggest that using a simple means–ends problem-solving strategy may retard learning of problem structure by novices. This problem structure, similar to what we are calling problem schemata, is discovered only if cognitive resources are available to notice correlations and patterns. Sweller and his coauthors suggest that these extra resources are not available during means–ends problem solving because all resources are being consumed by that process. The active condition of Experiment 1 can be thought of as forcing the problem solvers to attend to features of the problems rather than only trying to reduce the distance to the goal. Subjects are required to generate an explicit hypothesis about what operator will be appropriate to correctly solve the problem based on what problem features are present.

Although passive concept formation occurs under certain conditions in other studies (Brooks, 1978; Reber, 1976), it is unclear whether it would occur in this experiment. In the studies of Brooks (1978) and Reber (1976) subjects had no initial rules for processing the items for categorization and the categorical structure was often quite complex. In this study subjects already have a set of rules for solving geometry problems which were learned during a high school geometry course. They must discriminate when to apply these existing rules, whereas in the previous studies subjects needed to form an initial set of rules to begin with. As noted earlier, the ACT* discrimination theory applies to learning only when subjects are getting feedback about the successes and failures of operator applications.

In addition to examining the role of active hypothesis testing we ask a second question in this experiment: How important is the semantic or logical connection between correlated features and operators to learning those correlations? Specifically, would features that were relevant to, or necessary preconditions for, the operation with which they were correlated facilitate abstraction of operator schemata? Does it matter whether the surface features were necessary to solve the proof or were, in final analysis, incidental to the proof? The ACT* theory predicts no effect of this variable on the discrimination process.[3] The relevance of feature to the task was crossed with active hypothesis testing in this between-subjects design.

METHOD

Subjects

Ninety-three members of the Carnegie–Mellon community served as subjects. They were self-selected as being "comfortable doing simple geometry proofs" and received class credit or were paid $3 per hour for the 2-h experimental session. A bonus of $1 was given for good performance. Subjects were assigned one of four conditions with 22 people in the incidental-passive condition (condition 1), 20 in the necessary-passive condition (condition 2), 20 in the incidental-active hypothesis condition (condition 3), and 31 in the necessary-active hypothesis condition (condition 4). Groups of from 2 to 10 subjects were run at a time.

Note that these subjects should already have well worked out operator schemata. This raises a question about the possibility of further tuning an already well-tuned system. Presumably, the impact of feature–operator correlations on learning would be greatest with geometry naive high school subjects. Various constraints prevented the use of this population. Using a college population should theoretically make demonstrating the acquisition of operator schemata more difficult.

Materials

Subjects in all conditions solved 20 learning problems and gave judgments on 32 test problems. The learning problems were 20 simple triangle congruence problems which contained various mixtures of discriminating features. The solution strategy was always either the angle–side–angle (ASA) or the side–angle–side (SAS) strategy of proving triangle congruence. Each problem contained enough information so that only one of these strategies could be used. The learning problems were printed on separate sheets of paper, one per sheet. Test problems were 32 triangle congruence problems similar to the learning problems except that 16 contained enough information to solve the problem with either the ASA or the SAS strategy and 16 did not contain enough information to be solved by either strategy. Test problems were printed onto transparencies and projected on a screen for 5 s each. Discriminating features were built into both the diagrams and the given information of the learning and test problems. Separate problem sets were constructed for necessary and incidental feature conditions. In the necessary condition, the discriminating features were necessary parts of the proof. In the incidental condition, these features were present, but could *not* be used in the solution of the proof. Figure 3 illustrates a problem from the necessary condition while Fig. 4 illustrates a problem from the incidental condition.

Vertical angles and isosceles triangles were the diagram features for the necessary condition. The vertical angles were highlighted with bold lines, as in Fig. 3. Subjects were instructed that bold lines indicated collinear line segments. All subjects were comfortable with this convention. The equivalent base angles of the isosceles triangles were marked into the diagrams. If this feature appeared in Fig. 3, sides $\overline{B}$ and $\overline{C}$ would be the equal sides of the isosceles triangle and the vertices of angles $\angle 2$ and $\angle 1$ would be joined to form the base of the isosceles triangle.

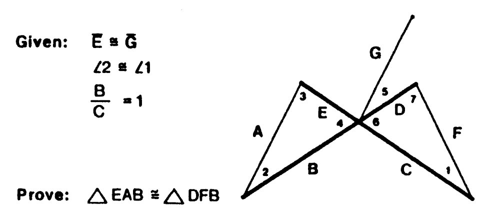

FIG. 3. Learning problem of mix 5 from the necessary condition of Experiment 1. Vertical angles and ratio of sides are the correlated features necessary to solve the proof using the ASA strategy. Bold lines indicate that line segments are collinear and hence the angles at their intersection are vertical angles. To simplify notation segments were referred to by single letters and angles by numbers.

[3] To assure that feature relevance was not confounded with simply noticing the feature, relevant and incidental features of roughly equal salience were selected, and subjects in all conditions were forced to process the features during problem solving. The details of these controls are enumerated in the Method section.

Given: $\overline{E} \cong \overline{D}$

$\angle 2$ comp. $\angle 5$

$m\angle 2 = 45$

$\overline{B} \cong \overline{C}$

$\angle 4 \cong \angle 6$

Prove: $\triangle EAB \cong \triangle DFB$

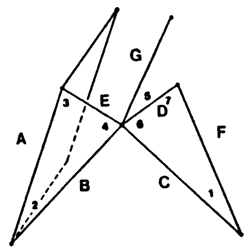

FIG. 4. Learning problem of mix 6 from the incidental condition of Experiment 1. The three-dimensional component of the diagram and the conplementary angles are the correlated features which are incidental to solve the proof using the SAS strategy. Note that it is not known if the pair of line segments E and C are collinear. Hence, it is not possible to infer that $\angle 4$ and $\angle 6$ are vertical angles, and therefore equivalent.

In the incidental condition the diagram features were a three-dimensional component drawn into the diagram and circumscription of the triangles. The three-dimensional component of a diagram has dashed lines to indicate an occluded line, as in Fig. 4. In both the learning and test problems, sides were labeled with, and referenced by, single letters. Angles were both labeled and referenced by single numbers. This convention is exemplified in Figs. 3 and 4. In both the incidental and necessary conditions these features were correlated with the choice of SAS or ASA. However, only in the necessary condition were these features involved in the final proof.

The features which appeared in the given information in both necessary and incidental conditions were ratio of side lengths (A/B = 1), as in Fig. 3, and a pair of statements involving complementary angles (measure of $\angle 1 = 45°$, $\angle 1$ complementary to $\angle 2$), as in Fig. 4. However, in the incidental condition these given features were not involved in the final proof solution. Pilot subjects ranked the diagram features of necessary and incidental problems as being of roughly equivalent salience when asked to describe the problem sets.

Features were paired and each pair was predictive of one solution strategy. In the necessary condition, vertical angles were paired with the ratio of sides and an isosceles triangle was paired with complementary angles. Each pair was predictive of one strategy. In the incidental condition, circumscription of the diagram was paired with the ratio of sides and a three-dimensional diagram component was paired with complementary angles. Again, each pair was predictive of one strategy. A particular problem could have or not have each feature present; thus there are $2^4 = 16$ possible problems, as indicated in Table 3. As a counterbalancing measure, two sets of problems were constructed for each cell of the design. In one set ASA was strategy A and SAS was strategy B, and in the other set this was reversed.

Twenty different problems, two each of mixtures 1–10 from Table 3, were used as learning problems. Problems of mixtures 1–16 were used as the test sets. Mixtures 1–4 are one-feature problems, mixes 5 and 6 are two-feature, or pure prototype, problems, and mixes 7–10 each have one feature that is predictive of one strategy and two features predictive of the other strategy. Mixes 7–10 contain features which conflict with one another in the predicted solution strategy. In the learning phase, the strategy which correctly solved the conflict problems (problems of mixes 7–10) was the strategy with the majority of correlated features present. Problems 11–16 contain no weight of features that support either strategy, and hence are referred to as ambiguous mixes. The given information in the learning problems contained the discriminating features (statements) and enough additional information to solve the problem using only the correlated strategy. Irrelevant given statements were

TABLE 3
Feature Mixtures in Problems Used in Experiment 1

	Feature mixture															
	1 Feature				Pure		Conflict				Ambiguous					
Problem feature	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Circumscription or vertical angles	A	0	0	0	A	0	A	A	A	0	A	0	A	0	A	0
3-D component or isosceles triangle	0	B	0	0	0	B	B	0	B	B	B	0	0	B	B	0
Ratio of sides = 1	0	0	A	0	A	0	A	A	0	A	0	A	0	A	A	0
Complementary angles, each equal to 45°	0	0	0	B	0	B	0	B	B	B	0	B	B	0	B	0
	Learning items (1–10)															
	Post-test items (1–16)															

Note. A denotes features predictive of strategy A; B denotes features predictive of strategy B; 0 denotes absence of that feature.

added to roughly balance the total number of references to angles and sides in the given information.

Two sets of 16 post-test problems were created. A set consisted of one instance of each of the feature mixes 1–16 from Table 3, presented in random order. The set for the first post-test contained the features specified in the feature mix and enough other features to solve the proof with either the ASA or SAS strategy. The second post-test set contained only the features specified in the mix. None of the problems in the second post-test were solvable by either strategy with the information given.

Manipulating Learner Strategy

In an attempt to vary active hypothesis testing, subjects in the different conditions solved the learning problems under different constraints. In the passive condition subjects simply solved the problems, writing the strategy and proof statements for a problem directly on the problem sheet. The problems were all simple enough that subjects solved them by marking the congruent sides and angles, noting whether SAS or ASA was applied by examination, and writing in the strategy and steps of the simple proof. They never had to venture a hypothesis about whether SAS or ASA would be correct before they had complete information. Thus, they never made errors from which they could learn.

In the active condition subjects interacted with a computer program which required explicitly selecting the strategy (ASA vs SAS) before solving the problems. The 20 problems to be solved were presented to the subject on separate pieces of paper, one per page. Problems in the active condition were identical to those in the passive condition (like those in Figs. 3 and 4), with one difference: They were missing a critical piece of information from the givens, which was necessary to solve the proof.

To solve a problem in the active condition subjects studied the problems sheet and typed in the problem number followed by a hypothesis about the strategy (ASA or SAS) that would correctly solve the problem. They then requested from the system the critical missing piece of information which was necessary to solve the proof using their hypothesized strategy. If the requested piece of information was the correct one to make the correlated strategy complete, the subject received points and continued to the next problem. If they were wrong, they lost points and were forced to choose a strategy again, requesting a new piece of information. This cycle of hypothesis and information request continued until the correct hypothesis and information was chosen. In nearly all cases this took only two attempts. The system presented prompts for the number of the problem being solved, for what strategy was guessed, and for what missing piece of information was requested. All prompts were presented and responses recorded via CRT terminal controlled by a DEC PDP11/34 computer.

Procedure

The first 10 min of the session were spent reading through a review of basic concepts in geometry and examples of other proofs. This review was constructed from relevant sections of a teacher's edition of Jurgensen et al. (1975). Subjects were told that they would be solving simple geometry proofs similar to those they had solved in high school. Their task was to become expert at solving the problems in the set that they were about to receive. Two simple proofs were solved by the experimenter to demonstrate the level of detail desired in the solution.

Following these instructions, subjects received a packet containing the 20 learning problems for their condition on separate sheets, randomly ordered. Subjects in the active condition were taken to individual experimental rooms, each containing a computer terminal. The interaction protocol with the computer was explained and demonstrated. Subjects were told

> There are features in the givens and the diagrams which might be helpful in predicting which strategy will be appropriate to solve a problem. Your task is to become an expert at solving these problems and predicting the correct solution strategy. It is possible to perfectly predict the solution to every problem, with experience.

Subjects in the active condition earned points for correct predictions and lost points for incorrect predictions. Subjects in the passive condition received their packet of 20 learning problems and were instructed how to record their answers on answer sheets. Both groups then spent 1 h in the problem-solving phase. All problems could be solved only one way, and all subjects completed all problems correctly.

Following completion of the problems, subjects in both conditions were instructed to sort their completed problems into stacks by correct solution strategy. If subjects finished early, they were instructed to review their solutions. To ensure that subjects processed the incidental features, specific instructions directed them to mark all the information in the given statements and inferences from the givens and diagram features on the diagram before solving the problem. Compliance was checked during the problem solving, and debriefing protocols confirm that subjects were aware of the incidental features, although not necessarily of their predictive value. After the subjects had completed the problem solving, sorted, and reviewed their strategies, the experimenter collected the problems and the test phase began.

In the test phase subjects made strategy judgments (ASA or SAS) for two sets of 16 briefly presented post-test problems. Presentation order of the post-test problems was randomized for each group of subjects run. Subjects were told that they would be presented with problems similar to those they had just solved and would have to pick a solution strategy for these new problems. Problems were presented via overhead projector on a screen at the front of the room. Subjects were instructed that the problems would be presented for only 5 s each and that they were not expected to be able to "logically" solve the problems in such brief time. It was suggested that they simply scan the problem features and, based on the problems they had just solved, give their "best guess" as to which strategy (ASA or SAS) would be used to solve the problems.

After subjects signaled readiness, they were presented a member of the test set for 5 s. They then recorded their strategy for the presented problem. After completing both test sets, subjects answered a set of debriefing questions intended to assess the strategies used to solve the test problems and to determine if the discriminating features were noticed.

RESULTS AND DISCUSSION

The data from the two post-tests was analyzed to assess how successful subjects were at learning the feature–operator correlations. If the subject's strategy response to a test problem was the same as strategy correlated with the majority of features in that problem, then it was scored as a correct strategy choice. If the strategy choice was different from that correlated with the majority of the features in the test problem, then it was scored as an incorrect strategy choice.

Of the 16 items in each post-test, 10 (mixes 1–10) had a majority of their features correlated with one of the two strategies. The mean numbers of correct strategy choices for these problems are presented in Table 4 classified according to the two variables—active vs passive-learning condition and incidental vs necessary features. A score of 5 would be

TABLE 4
Mean Number of Times Subjects Chose the Correlated Solution Strategy in the First Post-test (T1) and the Second Post-test (T2) in Experiment 1

Learning strategy	Incidental		Necessary	
	Cond 1		Cond 2	
	T1	T2	T1	T2
Passive	$\bar{x}$ = 5.36	4.86	$\bar{x}$ = 5.60	5.05
	SD = 0.36	0.32	SD = 0.35	0.34
	N = 22		N = 20	
	Cond 3		Cond 4	
	T1	T2	T1	T2
Active hypothesis testing	$\bar{x}$ = 5.90	5.90	$\bar{x}$ = 6.55	6.39
	SD = 0.42	0.45	SD = 0.32	0.40
	N = 20		N = 31	

(Relation of feature to proof solution: Incidental / Necessary)

Note. A score of 5 would be random performance; SD = standard deviation of the mean.

random choice of operators, a score of 10 would be perfect performance. A sign test was applied to determine whether subjects performed significantly better than chance (more than 10 out of 20 items correct over the two tests). Both of the active hypothesis testing conditions were significantly different from chance by a sign test. By the same test, neither of the passive conditions were significantly different from chance.

A three-way analysis of variance using post-test (first or second), incidental versus necessary features, and active versus passive hypothesis testing as the factors was performed on post-test accuracy scores. The only significant variable was active versus passive hypothesis testing, $F(1,89) = 9.54$, $p < .01$. Main effects of feature necessity, $F(1,89) = 1.56$, $p > .05$, and post-test, $F(1,89) = 1.76$, $p > .05$, were not significant, nor were interactions. The ACT* theory is consistent with both the significant effect of active versus passive hypothesis testing and the lack of a significant effect of incidental versus necessary features.

The results of both analyses suggest that subjects only learned the feature–strategy correlations in the active condition. Closer examination of the pattern of results in this condition provides more evidence that the results resemble those found in concept formation.

The accuracy of the 51 subjects in the active condition was examined for the three types of test problems:

- One-feature items: contain one feature correlated with the correct strategy (mixes 1–4 in Table 3).
- Pure prototype items: contain both features correlated with the correct solution strategy (mixes 5 and 6 in Table 3).

- Conflict items: contain two features correlated with the correct solution strategy and one feature correlated with the incorrect strategy (mixes 7–10 in Table 3).

Active condition subjects chose the correct operator for 59% of the conflict items, 60.5% of one-feature items, and 72.5% of the pure prototype items. The difference between the conflict item and the one-feature item accuracies is not significant by t test ($t(49) = 0.21$, $p > .05$), but performance on the pure prototype items is significantly superior to performance on the conflict items ($t(49) = 2.26$, $p < .05$) and the one-feature items ($t(49) = 2.21$, $p < .05$). This suggests that accuracy is a function of the absolute difference between correct and incorrect features. Interestingly, the pure prototype item accuracy is about twice as high above chance (50%) as are the conflict and the one-feature item accuracies.

As described earlier, ACT* theory views the discrimination process as automatic: Successful discrimination learning does not depend on the subject consciously identifying the contingencies. In the postexperiment questionnaire subjects were asked what the correlations were between features and proof strategies. Subjects in the active condition varied considerably in their ability to report the contingencies. The 51 subjects in the active condition were divided into 31 who accurately reported at least one feature–operator contingency (accurate report group) and 20 who failed (inaccurate/nonreport group). These groups were distributed evenly across the two active conditions. The accurate report group chose the correct strategy in 75% of the test problems. The inaccurate/nonreport group chose the correct strategy in only 49% of the test problems. This is equivalent to random performance by the inaccurate/nonreport group and strongly suggests that conscious identification of the contingencies is important in learning these simple rules. This contrasts with the ACT* theory which allowed for automatic and unconscious learning.

Thus, results imply that subjects will pick up on arbitrary feature–strategy correlations, but only if they are actively choosing that strategy against an alternate one. This contrasts with the prototype formation results found by Brooks (1978) and Reber (1976). Subjects in those experiments simply had to observe features–category correlations of items from poorly defined categories in order to perform better than subjects actively trying to discriminate the categorization rule. Since there is a small, but nonsignificant, positive effect on accuracy in the passive learning condition it is possible that with many more presentations of stimuli a significant effect might appear.

A prototype effect like that found in other concept formation and classification learning studies also appeared. That is, subjects are more likely to apply an operator the closer the situation is to the prototype for that operator. The next experiment pursues this issues further.

In summary, this experiment turned up results consistent with the ACT* theory of discrimination learning except for the issue of automatic learning, where the predictions of the theory are disconfirmed. The implications of this pattern of confirmation and disconfirmation of ACT* predictions are addressed again later.

EXPERIMENT 2

A prototype effect was found with the materials in Experiment 1: The more prototypical features a problem contained, the better subjects were at solving it. However, when the prototype only has two features it is hard to perform any incisive tests of this prototype effect. For instance, is it any different than what would be predicted from other theories of discrimination learning, concept identification learning, and schema abstraction? To begin to pull apart theories, operators with more complex feature sets needed to be examined. Our second experiment investigated two issues. First, it explored how problem-solving operators are learned in situations with more features and more applicable operators. Would prototype results similar to those found in Experiment 1 appear with more complex feature sets? Second, we introduced an instructional independent variable; filled delay of feedback regarding the correctness of operator choice. This delay of feedback filled with an intervening task, shown in the literature to be influential in simple learning tasks (Bourne, 1966), could also have strong effects on acquisition of complex skills. Following our presentation of the new, more complex, learning domain's design, we present our explanation of and rationale for how delay of feedback was varied.

It is difficult to manipulate problem features in a domain like geometry in a way that avoids confounding with natural variables of the domain. Having established that operator abstraction can occur in geometry, we constructed an artificial problem-solving domain based on the structure of the geometry proof domain used in Experiment 1. This new domain was more amenable to experimental manipulation. Couched in terms of a maze or dungeon search game, this artificial domain allowed little, if any, transfer from previously learned semantics due to the arbitrary combination of predictive features with operators. This permitted freedom to counterbalance the design and controlled for prior knowledge. Search through the rooms of the maze could only take place by data-driven forward search, without the aid of sophisticated planning behavior. There was no interroom structure in the domain that could be learned and possibly utilized to improve search performance. However, we emphasize

that, at an abstract level, deciding which move to make in a room is no different from deciding what postulate to apply in a geometric proof. Thus, the same issues can be addressed in this artifical domain.

The game involved progressing through a series of rooms. Each room consisted of a set of features (for instance, a fireplace, spiders) where various operators could be applied. Each operator transported the subject to a new room with a new set of features. Some rooms were dead ends from which the subject had to back out. The subject's task was to find the sequence of rooms which led him/her to the treasure room. Thus, he/she had to learn which operators in a given room navigated along the correct path and which led to dead ends. The correct operator depended on the features of the rooms. Thus, a particular operator was only successful in rooms that contained certain sets of features. As in Experiment 1, success depended on learning the feature–operator correlations.

Features were also included which appeared only when the subject was beginning to search down a dead-end solution path. We wanted to see if subjects could learn features that indicated they should give up on a path and try some other.

In addition to varying the prototypicality of the larger feature sets, we also manipulated feedback delay. The issue of feedback delay is interesting because it gets at a common issue in problem solving. Often the student does not learn that a choice is incorrect until later when the problem is completed. He/she must then propagate this information back to where the decision was made. How much impact, if any, do such delays have on learning? The ACT* theory predicts an effect of delay because the discrimination process only selects from features present in working memory when feedback is given. With a delay there is a danger that the critical information will not be in working memory. However, there are possible benefits from making errors. A student might notice features which indicate that the current line of problem solving is nonoptimal or cannot lead to a solution. These features could be compiled into a production which would reduce erroneous problem-solving attempts.

In the immediate feedback condition, when the subject applied an inappropriate operator it was immediately evident: The operator failed to transport the subject to a new room in the game, a message appeared stating that the player had encountered a dead end, and points were subtracted for the error. Subjects in the delayed feedback condition were allowed to enter the incorrect room. Initially, delayed feedback subjects would realize that they had previously picked an inappropriate operator only after attempting to apply another forward operator in that new, incorrect-path room. That operator would fail to transport them to a new room and they would receive the same feedback that was received by subjects in the immediate feeback condition. The only difference was that the delay feedback subjects received the feedback that a previous choice was incorrect while one move down an incorrect path. The immediate feedback condition models a very constrained teaching situation. The student is immediately informed if he/she has made an error. The delayed feedback condition models a less constrained teaching situation. The subject is allowed to discover his/her mistakes further into the problem solving before getting feedback from the teacher. A third level of delay was originally included, with two levels of incorrect search allowed before encountering dead ends. This proved to be too time consuming for subjects to finish. They characteristically got lost and spent too much time wandering around in the incorrect parts of the maze.

The experiment took place in a single session consisting of four phases. An instruction and introduction phase was followed by the operator memorization phase. Subjects then spent from 1 to 1.5 h in the game, or learning phase, followed by a post-test phase. The subjects were instructed that through experience with the maze they could learn which of the operators would be correct at any given point in the maze.

METHOD

Subjects

Twenty members of the Carnegie–Mellon University community served as subjects—ten in the immediate and ten in the delay condition. Pay was $3 per hour of participation. Subjects took an average of 2 h in the immediate feedback condition and 2.5 h in the delayed feedback condition. A bonus of $1 was given for good performance, as assessed by the number of points accumulated in the game portion of the experiment.

Materials

All stimuli were presented, and reponses recorded, via CRT terminal controlled by a DEC PDP11/34 computer. After an introduction, subjects memorized the condition–operator pairs. Each pair was presented as a conditional rule relating the features to an action. For example, Action 1 was, "If the room has a fireplace and a roomkeeper, then bow to the roomkeeper." The seven condition–operator pairs are listed in Table 5 as they were presented to the subject. Pairings of features to operator were the same for all subjects. The function of these fixed feature–operator pairs is described later. Subjects typed the number of the operator to communicate their choice to the program which ran the experiment. The letter *b* was typed to communicate choice of the backup operator. These operators can be thought of as general rules analogous to the general postulates of geometry.

Subjects moved from room to room (a room being a description of an indoor scene, to be explained shortly) by selecting one of the memorized operators while in the room. At each room there were three possible operators: two possible forward actions and the backup action. Choosing the backup operator re-presented the prior room visited, as if the subject backed up into the previous room. The decision between these three options was based on the features present in the described room.

Each room has two sets of features: a set of three fixed features and a set of four discriminating features. The set of three fixed features, displayed at the top of the room description (see Fig. 5), combined to form the conditions for the two possible forward

TABLE 5
Fixed Feature–Operator Pairs Used in Experiments 2 and 3

Feature pairs	Associated operator
Fireplace Roomkeeper	Bow to the roomkeeper (1)
Roomkeeper Door	Knock on the door (2)
Door Wand	Wave the wand (3)
Wand Stairs	Take the stairs (4)
Stairs Tunnel	Enter tunnel (5)
Tunnel Lamp	Rub the lamp (6)
Lamp Fireplace	Light the fire (7)

Note. The number following the action is what the subjects typed to communicate to the program which operator they wished to apply. The pairings of fixed feature pair to operator were the same for all subjects. The fixed features appeared in random order in the room descriptions.

operators for that room. These features were elements of the conditions in the memorized condition–operator pairs. In Fig. 5 the two operator choices for the mix of fixed features are 1 (bow) and 2 (knock). The set of fixed features prescribes only the two possible choices and gives no information about which of the two might be the correct choice.

The other set of four features presented in Fig. 5 are discriminating features. They contained information about the correct operator choice in the room. Each subject received a random assignment of discriminating feature-to-operator correlations. Discriminating features described the noise heard, inhabitants, atmosphere, and other objects in the room. These discriminating features appeared in several different mixtures. The mixtures varied the predictive content in different rooms. Descriptions and rationale for the different mixtures of discriminating features are presented in the following section.

There is a fairly direct mapping between the structure of this experiment and the structure of the geometry experiment. Consider the choice in Experiment 1 to try the side-angle-side postulate versus the angle-side-angle postulate in the presence of given information about a side and an angle congruence. The *side* and *angle* congruence were the fixed features in that experiment that dictated the operator choice. The other diagram features that correlated with operator choice were the discriminating features. In both experiments, subjects start out knowing the connection between the fixed features and the operator; they have to learn the probabilistic connection between discriminating features and the operator choice.

Mixtures of Discriminating Features

Each of the seven operators had a set of four discriminating features (room attributes) associated with it. These four features made up the prototypical room associated with a particular operator. There were also eight features associated with the backup operator and

You are in a room which contains

a brick fireplace
a foul-smelling troll roomkeeper
a polished brass door

You hear	wicked laughter
The room also contains	hairy spiders
The room is	musty and dusty
You also notice	burning torches

... What do you want to do?

FIG. 5. Sample room display from Experiment 2, with three fixed features at the top and four discriminating features at the bottom.

eight not associated with any operator (neutral features). The neutral features gave no information about correct operator selection. As mentioned earlier, sets of discriminating features were formed for each subject. All features were randomly drawn from the same pool of features and sets were randomly assigned to operators.

These different sets of features were combined to form mixtures of features with varying amounts of information concerning the correct operator choice in any given room. The mixtures are presented in Table 6. During the game phase, while on the correct solution path, the subject encountered only mix 1 (two positive features associated with the correct operator, one negative feature associated with the incorrect operator, and one neutral), mix 2 (three positive, one negative), mix 3 (two positive, two neutral), and mix 4 (three positive, one neutral). Subjects in the immediate feedback condition never saw the partial dead-end mixture, mix 5 (one negative feature for each of the two forward operators and two backup features), since it appeared only in room descriptions which were off of the correct solution path. The pure dead-end feature mix, mix 6 (four backup features), and pure prototype mixture, mix 7 (all four features for the operator), were never seen in the game phase, but appeared in the post-test phase. The correct response to the two dead-end feature mixtures in the post-test (mixes 5 and 6) was *b* for backup.

TABLE 6
Feature Mixtures in Stimuli Used in Experiments 2 and 3

Correlated features	Feature mixtures 1	2	3	4	5	6	7
Correct operator	2	3	2	3	1	0	4
Incorrect operator	1	1	0	0	1	0	0
Neutral (no operator)	1	0	2	1	0	0	0
Dead end (backup operator)	0	0	0	0	2	4	0
Immediate feedback condition	Learning items						
Delayed feedback condition	Learning items						
Both feedback conditions	Post-test items						

The room descriptions were designed such that each operator choice is correct in two different contexts, as defined by the three fixed features. That is, each operator required two fixed features, A and B. By adding C or D, two contexts could be created, each with a different competing operator.[4] The seven different operators thus produced 14 different choice situations. Each of these 14 choice situations occurred eight times over the course of the experiment, with each of the four mixtures occurring twice. This leads to 112 decisions which must be made correctly in both the immediate and the delay feedback conditions in order to complete the task. The decisions were grouped into "games" of four decisions each. After four correct decisions, subjects reached the treasure room, received points for the treasure found, and were presented with their current point total. Subjects played 28 games to see all 112 combinations.

Post-test items were constructed by combining each of the seven mixtures for an operator with each of the 14 different action decisions (seven operators in two different contexts) yielding 98 post-test items for each subject.

Procedure

Subjects began the introduction phase by reading a description of the game's objectives. The stated goal of the game was to become expert at moving through the mazes, i.e., selecting only operators that moved along the correct path to the treasure rooms. As mazes were completed, points were accrued. As errors were made, points were lost. The structure of the mazes was described as a series of rooms, each with one entrance and two possible exits. Subjects were informed that the features in the room description defined the two legal operators which could be used for transition through the exists and that the *b* key would back them up to the previous room. The instructions explicitly stated that, through experience, information concerning which of the operators would be correct, in any given room, could be derived from certain features in that room.

Following an opportunity for subjects to ask questions, the memorization phase began. The subject memorized the set of legal operators and the features which defined when they could be used. When the subject felt that he/she had the seven condition–operator pairs fairly well memorized (using any method or technique they wished), a double dropout test on the pairs was given. Each pair was presented twice, once with the conditions in the originally presented order, and once in reversed order. The 14 pairs were presented in random order. Requiring this initial memorization of the condition–operator pairs was based on two considerations. First, this requirement enforces the constraint that every subject came into the learning phase of the experiment with the same minimum level of learning of the operators. Second, theorems in the geometry domain are traditionally taught in some rote form before they are used. Upon completion of the double dropout test, the subject entered the game phase.

The game phase involved the subject moving through 28 games of four moves each, making decisions at every room, with feedback concerning their current point total at the end of every game. At each choice point (room) the selected operator, reaction time of the operator choice, number of previous visits to that room, and other relevant data were recorded for later analysis.

The post-test phase began after the subject finished the 28th game. Room descriptions were presented exactly as in the game phase, but the rooms were not connected to one another. The subject's task was to choose the correct operator for each of the isolated

[4] For instance, in Table 5 the two features *roomkeeper* and *door* call for *knock on the door*. By combining these two features with *fireplace*, the operator *bow to the roomkeeper* is also possible. By combining the two feature with *wand*, the operator *wave the wand* is also possible.

rooms presented. Their operator choice was recorded for each presented room description. No feedback was given regarding the correctness of the operator selection.

RESULTS AND DISCUSSION

Table 7 presents the mean accuracy scores and percentage correct from Experiment 2 classified by feature mixture (mixes 1–7, listing the number of constituent features correlated with different operators) and condition (immediate vs delayed feedback). Incorrectly choosing the backup operator for mixes 1–4 and 7 accounted for 8.3% of the errors in the delay condition. Subjects in the immediate feedback condition never had to back up, and consequently never used "backup" as a response in the post-tests. The backup errors were excluded from the analysis of variance post-tests. The backup errors were excluded from the analysis of variance computations in order to compare performance on erroneous forward moves only.

We performed a three-way analysis of variance on the accuracy scores from mixes 1–4 and 7. The factors were feature mixture, operator, and feedback delay. The only significant main effect found was due to mixture, $F(4, 72) = 6.78$, $p < .0001$. Post-test accuracy between immediate and delayed feedback conditions showed no significant differences, $F(1, 18) = 0.81$, $p > .05$. The overall post-test accuracy for mixes 1–4 and 7 was 65.3% for the immediate feedback condition, while it was 70.6% for the same mixes in the one-delay condition. There were no significant interactions.

Performance on mixes 5 and 6 for the delay group shows low accuracy with high standard deviations. If delay subjects are partitioned into those that never used the backup response during the post-tests and those who did, an interesting result emerges. The mean percentages for the delay group subjects who used the backup response (six subjects) for mixes 5 and 6 were 61.9 and 85.7%, respectively. Random performance would yield 33.3% accuracy. This finding implies that subjects in the delay condition were very good at detecting situations where they should stop following the current line of search and back up.

Significantly better performance on feature mixes with more correlated features supports the hypothesis that prototype exemplars are learned more accurately than nonprototypic exemplars, as found in Experiment 1. Given 16 exposures to different mixes of features, the percentage accuracy of the performance on test items is similar to that found in the active/necessary condition of Experiment 1.

We were somewhat surprised to find no significant effect of feedback delay in Experiment 2. The general hypothesis is that delay subjects are less likely to associate features with operators. The effect is cumulative. If p is the difference in probability of association between the immediate and delay conditions, and the number of opportunities for association is

TABLE 7
Post-test Accuracy Means, Standard Deviation of the Mean, and Percentage Correct by Feature Fixture from Experiment 2

	Number of features				Feedback condition					
					Immediate $n = 10$			Delay $n = 10$		
Mix	Pos	Neg	Neut	Dead end	$\bar{x}$	*SD*	% Correct	$\bar{x}$	*SD*	% Correct
1	2	1	1	0	8.4	(0.70)	60.0	8.4	(0.78)	60.0
2	3	1	0	0	8.2	(0.57)	58.6	9.8	(0.81)	70.0
3	2	0	2	0	9.0	(0.68)	64.2	9.2	(1.00)	65.7
4	3	0	1	0	9.5	(0.73)	67.9	10.3	(0.80)	73.6
5	1	1	0	2	0	(0)	0	5.2	(1.81)	37.1
6	0	0	0	4	0	(0)	0	7.2	(2.03)	51.4
7	4	0	0	0	10.6	(0.96)	75.7	11.7	(0.63)	83.6

Note. Mean scores represent performance out of 14 possible correct.

n per day, there is a difference of *pn* associations after Day 1. Experiment 3 was performed to explore the possibility that the effect of delay was small and incremental. If such an effect existed it might produce significant differences in learning accuracy with extended practice: After Day 2 there would be a difference of 2 *pn*. Extended practice also would provide a larger corpus of data for quantitative comparisons of the process model in this study to other models of schema abstraction.

EXPERIMENT 3

Experiment 3 was carried out for two purposes: to investigate whether any small and incremental effects of feedback delay might emerge with extended problem-solving experience and to gather more data for comparison with two models of the task. Experiment 3 is simply an extended version of Experiment 2, doubling the experience with the task across 2 days. The only major addition was a pretest before the second day's session. All other details of the experiment are identical to Experiment 2.

METHOD

Subjects

Subjects were 25 members of the Carnegie–Mellon University community. They participated under the same conditions as in Experiment 2, with 13 subjects in the immediate condition and 12 in the delay condition. Sessions lasted from 2 to 2½ h, on 2 consecutive days.

Materials and Procedure

Materials were identical to those used in Experiment 2. The correlation of features with operators was random for each subject, but consistent across days of the study. Subjects worked through 28 games of four correct operator choices per game on the first day, as in Experiment 2. On the second day they again worked through another series of 28 games, using the same feature correlations. This gave subjects 32 presentations of each operator across days (four mixes of features occurring four times daily for each operator). Following both Day 1 and Day 2 learning phases, subjects were given a post-test. Also a pretest preceded the Day 2 learning phase. Pretest items were generated exactly in the same manner as post-test items. The pretest was meant to be an assessment of how much recall subjects were bringing to the second session.

Subjects proceeded on both days exactly as in Experiment 2, with the exception of the pretest before the second session in Experiment 3. Before the second session subjects were told that they would be continuing in a maze like the one they had been in the previous day, and that everything they might have learned in that session would still be valid.

RESULTS AND DISCUSSION

The post-test accuracy means and percentage correct for post-tests on Days 1 and 2 are presented in Table 8. Incorrectly choosing the backup operator for mixes 1–4 and 7 accounted for 16.4% of the Day 1 errors

and only 4.8% of the Day 2 erros in the delay condition. Subjects in the immediate feedback condition never had to back up, and consequently never used "backup" as a response in the post-tests. The backup errors were excluded from the analysis of variance computations in order to compare performance on erroneous forward moves only. Thus, the analysis of variance was performed on percentage correct of the forward moves.

The primary result of Experiment 3 was that extended practice had an effect on accuracy between delay conditions. There was no significant difference between accuracy in the immediate and delay conditions after the first day of practice, $F(1, 22) = 3.31$, $p > .05$. After the second day of practice subjects in the immediate feedback condition were significantly more accurate than those in the delay conditions, $F(1, 22) = 4.53$, $p < .05$. There was also a main effect for feature mixture, as in Experiments 1 and 2, $F(4, 88) = 13.82$, $p < .0001$. No interactions proved to be significant.

As in Experiment 2, performance on mixes 5 and 6 for the delay group on both days shows low accuracy with high standard deviations. If delay subjects are partitioned into those who never used the backup response during the post-tests and those who did, a result very similar to that found in Experiment 2 emerges: The mean percentages for the group which used the backup response for mixes 5 and 6 were 56.4 and 90.0%, respectively, on Day 1 (seven subjects) and 68.6 and 90.7%, respectively, on Day 2 (nine subjects). Random performance would yield 33.3% accuracy. This finding implies that subjects in the delay condition were very good, even on Day 1, at detecting situations where they should stop following the current line of search and back up.

Significant differences also appeared in the pretest data analysis. The accuracy means and percentage correct for the pretest on Day 2 are presented in Table 9. We performed a three-way analysis of variance on the accuracy scores of the pretest from mixes 1–4 and 7. The factors were feature mixture, operator, and feedback delay. Significant main effects were found for mixture, $F(4, 72) = 5.31$, $p < .001$, and feedback condition, $F(1, 18) = 5.82$, $p < .05$. The overall post-test accuracy for mixes 1–4 and 7 was 53.4% for the immediate feedback condition, while it was 44.7% for the same mixes in the one-delay condition. There were no significant interactions. Subjects are below 50% in the delay condition because of intrusions of erroneous backup responses.

The negative effect on forward move accuracy in the delay condition of the post-test is also consistent with the predictions of ACT* theory. The relevant features for discrimination are not in working memory when the correct forward move is chosen after backing up into a room in which the wrong move was previously chosen. This could be due to forgetting

TABLE 8
Post-test Accuracy Means, Standard Deviation of the Mean, and Percentage Correct by Feature Mixture for Both the First and Second Day Post-test from Experiment 3

	Number of features				Feedback condition					
					Immediate $n = 13$			Delay $n = 12$		
Mix	Pos	Neg	Neut	Dead end	$\bar{x}$	*SD*	% Correct	$\bar{x}$	*SD*	% Correct
					Day 1					
1	2	1	1	0	8.1	(0.52)	57.7	7.8	(0.82)	55.4
2	3	1	0	0	9.5	(0.50)	68.1	7.8	(0.59)	55.4
3	2	0	2	0	10.3	(0.58)	73.6	9.1	(0.70)	64.9
4	3	0	1	0	11.8	(0.36)	84.6	9.4	(0.83)	67.3
5	1	1	0	2	0	(0)	0	4.7	(1.38)	33.3
6	0	0	0	4	0	(0)	0	7.3	(1.91)	52.4
7	4	0	0	0	11.9	(0.50)	85.2	10.2	(0.76)	72.6
					Day 2					
1	2	1	1	0	10.2	(0.67)	73.1	9.0	(0.71)	64.3
2	3	1	0	0	12.2	(0.50)	87.4	10.2	(0.71)	72.6
3	2	0	2	0	11.4	(0.56)	81.3	10.3	(0.82)	73.8
4	3	0	1	0	12.9	(0.39)	92.3	10.9	(0.83)	78.0
5	1	1	0	2	0	(0)	0	7.2	(1.65)	51.2
6	0	0	0	4	0	(0)	0	9.3	(1.75)	66.7
7	4	0	0	0	13.5	(0.21)	98.4	11.3	(1.03)	81.0

Note. Mean scores represent performance out of 14 possible correct.

TABLE 9
Pretest Accuracy Means, Standard Deviation of the Mean, and Percentage Correct by Feature Mixture for the Second Day Pretest from Experiment 3

	Number of features				Feedback condition					
					Immediate $n = 10$			Delay $n = 10$		
Mix	Pos	Neg	Neut	Dead end	$\bar{x}$	*SD*	% Correct	$\bar{x}$	*SD*	% Correct
1	2	1	1	0	6.6	(0.48)	47.1	5.7	(0.50)	40.7
2	3	1	0	0	7.7	(0.42)	55.0	5.8	(0.50)	41.4
3	2	0	2	0	7.0	(0.46)	50.0	6.0	(0.49)	42.9
4	3	0	1	0	7.9	(0.41)	56.4	6.7	(0.47)	47.9
5	1	1	0	2	0	(0)	0	3.1	(0.47)	22.1
6	0	0	0	4	0	(0)	0	5.4	(0.50)	38.6
7	4	0	0	0	8.2	(0.34)	58.6	7.1	(0.46)	50.7

Note. Mean scores represent performance out of 14 possible correct.

caused by the delay or interference from the intervening operator choices. This is also consistent with the reports of some subjects' behavior in the delay condition. After they had backed up into the previous room they simply chose the other operator (the one which they did not choose the first time) without reevaluating the features present in that room. However, the delay subjects did learn something that the immediate subjects did not: the features that indicate that a room is a dead end.

While this result of feedback delay decreasing accuracy needs to be replicated, it has important possible implications for choice of tutorial strategy and implementation of computer-based tutors (Anderson, Boyle, Farrell, & Reiser, 1984). An important issue in tutoring problem solving is whether to give feedback immediately when the student makes a wrong move or to wait and let the student discover his or her error by hitting a dead end. The results of this experiment have two implications for this issue: First, a student will learn correct moves faster if he/she is given immediate feedback—which does not correspond with everyones intuition. Second, the student will learn to recognize dead ends better if he/she is allowed to encounter them—which seems more intuitive. These issues are discussed at greater length in the conclusion section.

Modeling Experiment 3 as a Classification Task

A major claim of this paper is that abstraction of operator schemata is similar to concept formation or classification learning. Performance in Experiments 1–3 is qualitatively similar to previous concept-learning and categorization results: These similarities lend strength to this claim. Quantitative similarities of this study to an analogous classification study would strengthen the claim further. This is the motivation for developing a model of Experiment 3 as if it were a classification study. The stimuli actually seen by subjects in Experiment 3 are used as classification items for a mathematical model of classification learning. This models Experiment 3 as if subjects were learning to classify rooms into categories, instead of learning to pick the correct operator for each room. The goal of this modeling is to be able to compare the observed data from Experiment 3 to predictions that a model of classification learning would make for the exactly the same data.

An item-based mathematical model of classification learning, developed by Medin and Schaffer (1978), was chosen because it is both representative of a class of instance-based models, and it has done well in contrasts with a variety of different types of classification models. The essence of the model is that a preceding stimulus is retrieved from memory to classify a new stimulus. The new stimulus is categorized as being in the same category as the retrieval stimulus. The probability of retrieving a particular item is the ratio of its similarity to the test item

over the summed similarity of all past items. The probability of classifying a test item into a given category is a ratio of the summed similarities of the past items in that category over the summed similarity of items in all categories.

The similarity between two items is computed according to the following rules. A similarity value of 1 is assigned for each feature on which the items are identical and a similarity value (parameter to be estimated) less than 1 is assigned for each mismatched feature. The overall similarity measure between the two items is the product of these similarity values for individual features.

The probability of correctly classifying the item is the ratio of the summed similarity of the item to all previously presented items in the appropriate category, over the similarity of that item to all items, both correct and incorrect.[5] In the case where the subject is choosing between two categories, A and B, this probability for the immediate feedback condition is given in Eqs. (1) and (2).

$$p_{\text{correct choice A}} = \frac{\text{summed similarity to A}}{\text{summed similarity to A and B}} \tag{1}$$

or

$$p_{\text{correct choice A}} = \frac{\sum_{i \epsilon A} \prod o_{ij}}{\sum_{i \epsilon A \text{ or } B} \prod o_{ij}} \tag{2}$$

where A is the correct category and B is the incorrect category. In Eq. (2), $o_{ij} = 1$ if the jth feature of the ith item is the same as the test item feature and $o_{ij} = a_j < 1$ otherwise. In the delayed feedback condition it is necessary also to consider the similarity of the test stimulus to backup stimuli studied. The equation for the delayed feedback condition is given in Eq. (3):

$$p_{\text{correct choice A}} = \frac{\text{summed similarity to A}}{\text{summed similarity to A, B, and backup stimuli}}. \tag{3}$$

This model was used in an analysis of the post-test data from both Day 1 and Day 2 of Experiment 3. The similarity of each item in the post-test to every learning item that was relevant to the operator choice for that item was computed. This includes the re-presentations of rooms if the subject had to back up and go through a previously visited room. For example, consider a post-test item in the immediate feedback condition

[5] For a complete description of the model, see Medin and Schaffer, 1978.

which required the choice between operators 1 and 2, with 1 as the correct choice. (This choice will be referred to as (1,2), where the first number in the parentheses always represents the correct choice.) To find the probability of correctly classifying this item we computed the similarity of the test item to all the items where 1 was the correct operator, namely the (1,2) and (1,7) items. We also computed the similarity to all the items where 2 was the correct operator: the (2,1) and (2,3) items. The probability of correctly selecting the operator for the item then is given in Eq. (4):

$$p_{\text{correct choice 1}} = \frac{\text{summed similarity to all (1,2) and (1,7) stimuli}}{\text{summed similarity to all the (1,2) (1,7) (2,1) (2,3) stimuli}}. \tag{4}$$

These similarities were then averaged across operators and across subjects within a feedback condition. The assumption was made that all similarity parameters (i.e., the a_j) were equal for all features. The values of similarity parameters for the features were set by tuning the model to the best root mean squared error fit to the data. The values for the feature similarity parameters were estimated at .44 for the immediate and .28 for the delay conditions for the first-day model fits and .25 for both the immediate and the delay conditions for the second-day model fits. These parameter estimates imply that the confusability or similarity among different features decreases from Day 1 to Day 2—which seems intuitive. However, the difference on Day 1 implies that the features are more easily confused in the immediate condition—which does not seem intuitive. Despite this parameter difference, performance was worse in the delay condition due to erroneous backup responses.

The predicted and observed results are presented for both immediate and delay conditions in Fig. 6 for the first day's post-test and Fig. 7 for the second day's post-test data. Overall, the fit for the immediate and delay conditions is reasonably good. Quantitative comparisons of goodness-of-fit are discussed in a later section.

Modeling Experiment 3 with Hypothesis Testing

Based on retrospective protocols from several subjects in Experiment 3, we generated a second, rather different simulation of discrimination in this task. This model is similar to the hypothesis testing model proposed by Levine (1966). The model is based on a conscious generation and test of discriminating features with restricted replacement of features sampled. When presented with a new stimulus item, the simulation checks the features to see if any have been learned as predictors of either candidate operator for the stimulus. The number of features associated with

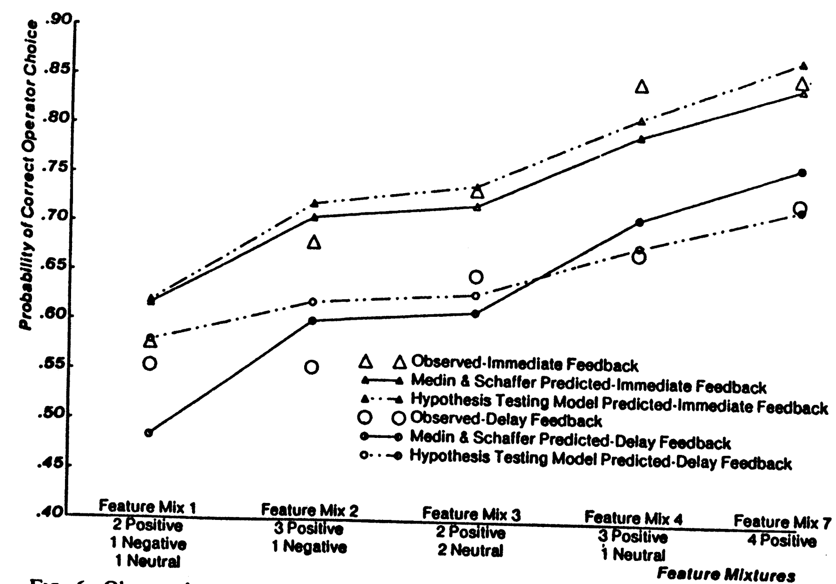

FIG. 6. Observed accuracy data from Day 1 of Experiment 3 and predicted data from two models.

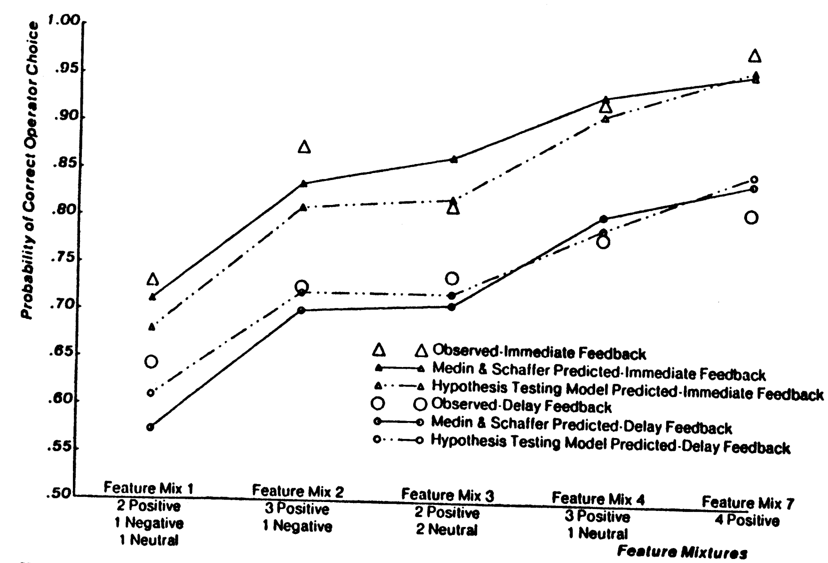

FIG. 7. Observed accuracy data from Day 2 of Experiment 3 and predicted data from two models.

each operator is summed and the operator with more "votes" is selected. If there is a tie between number of votes for each operator, or no features in the stimulus have been learned (as is the case early in the task), then the simulation randomly selects one of the two operators.

Feedback following the choice informs the simulation whether the choice was correct or incorrect. If the choice was correct, then nothing changes. The correlations of features to operators remain the same. If the feedback is negative, then the simulation picks any feature which is still unlearned (i.e., has no operator correlated with it). This unlearned feature is then learned as predictive of the appropriate operator with estimated probabilities of .525 in the immediate condition and .345 in the delay condition. These probabilities of successfully memorizing the correlation of a feature with an operator come from tuning the simulation to fit the observed data from Experiment 3.

If no unlearned features exist, then the simulation randomly selects a feature which is either learned as predictive of the wrong operator or is currently correlated with an operator which is not one of the two valid candidates for the given stimulus. That feature is then relearned as predictive of the appropriate operator, with a probability of success, again, of .525 and .345 for the immediate and delay conditions, respectively. The effect of randomly relearning feature–operator correlations is that it provides opportunity to change learned correlations that may have been erroneously learned.

There are two observations to note about these parameters. First, the probability is lower in the delayed condition to reflect the lower probability of putting the feature and operator together at a delay. Second, there is just one parameter for both Day 1 and Day 2. The higher performance on Day 2 is due to the greater opportunity to have formed the correct feature–operator correlations.

The results of the hypothesis testing model are also presented in Figs. 6 and 7. The fit of the hypothesis testing simulation to the data is quite good and similar to the results predicted by the Medin and Schaffer model. A discussion of the goodness-of-fit for these models follows.

Comparing the Models' Fits to the Data

Given the post hoc nature of the hypothesis testing model, we cannot make strong claims regarding its general predictive value. However, the difference in fit of the two models to the data may still be of interest. The predictions of both models depend on estimating a set of independent parameters. In the case of the Medin and Schaffer model, one parameter was estimated for each of the conditions on each day: Day 1-delay, Day 2-delay, Day 1-immediate, and Day 2-immediate. Just two parameters were estimated for the hypothesis testing model—one probability for the

delay and one for the immediate condition. In all cases the best fitting parameter was obtained by a simple grid search of parameter values, looking for the parameter that gave the best fit to that subset of the data which that parameter predicted. Predictions were obtained for the Medin and Schaffer model by mathematical calculation and for the hypothesis testing model by Monte Carlo simulation. The degrees of freedom for the models are the total number of observations, 20, minus the number of parameters estimated. So there are 16 degrees of freedom in the Medin and Schaffer model and 18 in the hypothesis testing model.

The χ^2 statistic was used to evaluate whether the two models differ in their predictive power. χ^2 values were computed for observed mean frequencies of correct and incorrect responses to the predicted mean frequencies for the those responses. The results, presented in Table 10, reveal that only the hypothesis testing model's predictions are not significantly different from the observed data. The total χ^2 for Medin and Schaffer is 36.82, while it is just 20.84 for the hypothesis testing model. This is despite the fact that the hypothesis testing model has more degrees of freedom.

Also note that the hypothesis testing model was able to predict the data across both days with single parameters, in contrast to the Medin and Schaffer model needing different parameters for each day. This is possible evidence that the effect of extra days is just to give more opportunities to form and test hypotheses which correctly associate features to operators. The major problem with the Medin and Schaffer model concerned its prediction of a steeper prototypicality function for the delay conditions than was obtained. In the Medin and Schaffer model, poorer performance is correlated with stronger prototypicality effects. In the data, just the opposite is the case. The observed difference between mix 7 and mix 1 in the data was 17% in the delay condition and 26% in the immediate condition. Medin and Schaffer predicted 24% difference in the delay condition and 22% in the immediate condition. The hypothesis-testing model predicts effects of 15% in delay and 24% in the immediate condition.

TABLE 10
Goodness-of-Fit Measures for Predicted to Observed Data for Experiment 3

	χ^2		Correlation (r)	
Comparison	Day 1	Day 2	Day 1	Day 2
Medin and Schaffer to observed	20.58	16.24	.922	.962
Hypothesis testing to observed	8.75	12.09	.974	.963
Hypothesis testing to Medin and Schaffer	—	—	.952	.986

The Medin and Schaffer model was not developed to account for abstraction of problem-solving schemata but rather of simple object concepts. It is an open question, beyond the scope of this paper, which model would do better if similar experiments were performed on concept learning. Recently, Dulany, Carlson, and Dewey (in press) have argued for conscious hypothesis testing when applied to one class of concept-learning experiments.

GENERAL DISCUSSION

We have argued that people form schemata for operators which are similar to the schemata they form for objects. To support this view three studies were performed which shed some light on several questions about the process of operator abstraction and its similarity to schema abstraction for objects.

A major result of the studies reported here is similar to many experiments studying concept formation: Prototypicality of exemplars generally predicts the accuracy of operator selection. All three studies found that the greater the amount of predictive information in a stimulus item, the greater the chance of accurately selecting the correct operator. Fitting models to the data from Experiment 3 revealed that the performance in the experiment could be well predicted by models of discrimination learning.

Many aspects of the ACT* theory of discrimination were confirmed, but it failed in its assumption of automatic discrimination. Rather, subjects had to be aware of their hypotheses to perform better than chance in Experiment 1. In Experiment 3, a deliberate hypothesis testing model fits results quite well. This model could be simulated by a set of productions in ACT* that formed and remembered declarative hypotheses which are open to conscious inspection. However, this is quite different from the ACT* discrimination mechanism that learns productions which are not open to conscious inspection. Anderson (1983b) has proposed a revision of ACT* in which discrimination is a conscious process operating on declarative memory. That report reviewed a series of results which called into question automatic production learning (discrimination or generalization) as envisioned in the ACT* theory or indeed any form of induction that was not strategy controlled. That paper proposed that more discriminate general productions could be acquired as a product of compiling the conscious process of forming declarative hypotheses. Thus, this research is one further piece of evidence promoting reassessment of the notion of automatic induction (see also Dulany et al., in press). In summary, the research in this paper supports the following conclusions:

1. People can learn the correlations between operators and surface features—i.e., people can learn operator schemata.

2. Operator schemata lead to prototypicality effects just as found with object schemata.
3. Operator schemata are only formed when subjects actually incorrectly apply their operators and get feedback as to their errors.
4. The learning of feature operator correlations is inhibited by delay of feedback filled with intervening problem solving.

This research has investigated variables which affect operator discrimination and tuning. This is also a major goal in education: How can educators improve the speed and accuracy of students who are learning to apply operators in problem solving?

The role of active hypothesis testing for learning simple operators seems clear. A student who is actively looking for correlations of features with solution strategies appears to learn the discriminating correlations more accurately than one who is passively observing the features of the problem. Other research suggests that the reason for acquiring operator schemata or learning problem structure is that resources are available to notice these patterns and correlations of problem structure to solution strategies. Keeping students actively involved while solving examples of what will be typical problems and pointing out the important features which predict solution strategy is a pedagogical practice which the findings reported here support.

The results of Experiment 3 imply that performance on forward move selection is better if practice applying operators occurs with immediate feedback. Delay of feedback lowered accuracy on forward move selection, but these subjects were also quite accurate at detecting when to back up. An operator which indicates when to give up on a solution path is desirable when solving challenging novel problems. Subjects in the immediate feedback condition, although significantly more accurate at selecting forward moves, might flounder if allowed to move off of the correct solution path. They lack the tuned backup operators learned by subjects who encountered the more complex delayed feedback domain.

REFERENCES

Anderson, J. R. (1980). *Cognitive psychology and its implications*. San Francisco: Freeman.

Anderson, J. R. (1981). Tuning of search of the problem space for geometry proofs. *Proceedings of the Seventh International Joint Conference on Artificial Intelligence* (pp. 165–170). Vancouver, British Columbia.

Anderson, J. R. (1982). Acquisition of cognitive skill. *Psychological Review*, 89(4), 369–406.

Anderson, J. R. (1983a). *The architecture of cognition*. Cambridge, MA: Harvard Univ. Press.

Anderson, J. R. (1983b). Knowledge compilation: The general learning mechanism. *Proceedings of the International Machine Learning Workshop* (pp. 203–212). Monticello, IL.

Anderson, J. R., Boyle, C. F., Farrell, R., & Reiser, B. (1984). Cognitive principles in the design of computer tutors. *Proceedings of the sixth annual conference of the Cognitive Science Society* (pp. 2–9). Boulder: Institute for Cognitive Science, University of Colorado.

Anderson, J. R., Greeno, J. G., Kline, P. J., & Neves, D. M. (1980). Acquisition of problem-solving skill. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition* (pp. 191–230). Hillsdale, NJ: Erlbaum.

Anderson, J. R., Kline, P. J., & Beasly, C. M. (1979). A general learning theory and its application to schema abstraction. In G. H. Bower (Ed.), *The psychology of learning and motivation* (Vol. 13, pp. 227–318). New York: Academic Press.

Bourne, L. E. (1966). Information feedback: Comments on Professor I. McD. Bilodeau's paper. In E. A. Bilodeau (Ed.), *Acquisition of skill* (pp. 297–313). New York: Academic Press.

Bower, G. H., & Trabasso, T. R. (1964). Concept identification. In R. C. Atkinson (Ed.), *Studies in mathematical psychology* (pp. 32–94). Stanford, CA: Stanford Univ. Press.

Brooks, L. (1978). Nonanalytic concept formation and memory for instances. In E. Rosch & B. B. Lloyd (Eds.), *Cognition and categorization* (pp. 169–211). Hillsdale, NJ: Erlbaum.

Bruner, J. S., Goodnow, J. J., & Austin, G. A. (1956). *A study of thinking*. New York: Wiley.

Chase, W. G., & Simon, H. A. (1973). The mind's eye in chess. In W. G. Chase (Ed.), *Visual information processing* (pp. 215–281). New York: Academic Press.

Chi, M. T. H., Feltovich, P. J., & Glaser, R. (1981). Categorization and representation of physics problems by experts and novices. *Cognitive Science*, 5(2), 121–152.

Davis, J. H., Carey, M. H., Foxman, P. N., & Tarr, D. B. (1968). Verbalization, experimenter presence, and problem solving. *Journal of Personality and Social Psychology*, 8(3), 299–302.

Dulany, D. E., Carlson, R. A., & Dewey, G. I. (in press). A case of syntactic learning and judgment: How conscious and how abstract? *Journal of Experimental Psychology: Human Learning and Memory*.

Elio, R., & Anderson, J. R. (1981). The effects of category generalizations and instance similarity on schema abstraction. *Journal of Experimental Psychology: Human Learning and Memory*, 7(6), 397–417.

Ericsson, K. A., & Simon, H. A. (1980). Verbal reports as data. *Psychological Review*, 87(3), 215–251.

Franks, J. J., & Brandsford, J. D. (1971). Abstraction of visual patterns. *Journal of Experimental Psychology*, 90(1), 65–74.

Gagné, R. M., & Smith, E. C. (1962). A study of the effects of verbalization on problem solving. *Journal of Experimental Psychology*, 63(1), 12–18.

Greeno, J. G. (1978). A study of problem solving. In R. Glaser (Ed.), *Advances in instructional psychology* (pp. 13–75). Hillsdale, NJ: Erlbaum.

Greeno, J. G. (1980). Some examples of cognitive task analysis with instructional implications. In R. E. Snow, P. Frederico, & W. E. Montague (Eds.), *Aptitude, learning, and instruction* (Vol. 2, pp. 1–21). Hillsdale, NJ: Erlbaum.

Hayes-Roth, B., & Hayes-Roth, F. (1977). Concept learning and the recognition and classification of exemplars. *Journal of Verbal Learning and Verbal Behavior*, 16, 321–338.

Hinsley, D. A., Hayes, J. R., & Simon, H. A. (1977). From words to equations: Meaning and representation in algebra word problems. In M. A. Just & P. A. Carpenter (Eds.), *Cognitive processes in comprehension* (pp. 62–68). Hillsdale, NJ: Erlbaum.

Hunt, E. B., Marin, J., & Stone, P. J. (1966). *Experiments in induction*. New York: Academic Press.

Jurgensen, R. C., Donnelly, A. J., Maier, J. E., & Rising, G. R. (1975). *Geometry*. Boston: Houghton Mifflin.

Larkin, J., McDermott, J., Simon, D. P., & Simon, H. A. (1980). Expert and novice performance in solving physics problems. *Science (Washington, D.C.)*, 208, 1335–1342.

Levine, M. (1966). Hypothesis behavior by humans during discrimination learning. *Journal of Experimental Psychology*, 71(3), 331–338.

Mayer, R. E. (1981). Frequency norms and structural analysis of algebra story problems into families, categories, and templates. *Instructional Science*, 10(2), 135–175.

Mayer, R. E. (1982). Memory for algebra story problems. *Journal of Educational Psychology*, 74(2), 199–216.

Medin, D. L., & Schaffer, M. M. (1978). Context theory of classification learning. *Psychological Review*, 85(3), 207–238.

Mervis, C. B., & Pani, J. R. (1980). Acquisition of basic object categories. *Cognitive Psychology*, 12, 496–522.

Neves, D. M., & Anderson, J. R. (1981). Knowledge compilation: Mechanisms for the automatization of cognitive skills. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition*. Hillsdale, NJ: Erlbaum.

Posner, M. I. (1969). Abstraction and the process of recognition. In G. H. Bower (Ed.), *The psychology of learning and motivation* (Vol. 3, pp. 44–100). New York: Academic Press.

Posner, M. I., & Keele, S. W. (1968). On the genesis of abstract ideas. *Journal of Experimental Psychology*, 77(3), 353–363.

Reber, A. S. (1976). Implicit learning of synthetic languages: The role of instructional set. *Journal of Experimental Psychology: Human Learning and Memory*, 2(1), 88–94.

Reed, S. K. (1972). Pattern recognition and categorization. *Cognitive Psychology*, 3, 382–407.

Rosch, E. (1978). Principles of categorization. In E. Rosch & B. B. Lloyd (eds.), *Cognition and categorization* (pp. 27–48). Hillsdale, NJ: Erlbaum.

Silver, E. A. (1981). Recall of mathematical problem information: Solving related problems. *Journal for Research in Mathematics Education*, 12(1), 54–64.

Sweller, J. (1983). Control mechanisms in problem solving. *Memory & Cognition*, 11(1), 32–40.

Sweller, J., & Levine, M. (1982). Effects of goal specificity on means–ends analysis and learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 8(5), 463–474.

Sweller, J., Mawer, R. F., & Ward, M. R. (1983). Development of expertise in mathematical problem solving. *Journal of Experimental Psychology: General* 112(4), 639–661.

Wilder, L., & Harvey, D. J. (1971). Overt and covert verbalization in problem solving. *Speech Monographs*, 38(3), 171–176.

(Accepted October 10, 1984)