On the Nature of Verbal Rules and Their Role in Problem Solving*

SETH CHAIKLIN

University of Pittsburgh

This paper develops the idea of a verbal rule as a concise, linguistic description of a plan for problem solving. The first section discusses the properties of verbal rules, including what they are, why they are useful, what kind of plan is described, the additional knowledge needed to convert them into performable procedures, and how they might be used in problem solving. The second section reports the results of an empirical study of the role of signed-arithmetic verbal rules when they were instructed as a problem-solving method. Analysis of problem-solving protocols yields three kinds of results. Mental procedures constructed from verbal rules can be structurally different from the instructed verbal rules. Difficulties arise in selecting and executing procedures because of inadequate parsing procedures. Verbal rules tend to be used optionally and drop out of performance when a person becomes skilled, but they can be recalled and used explicitly when appropriate mental procedures are unavailable. Drawing on the results of the empirical study, the third section discusses the role of verbal rules in learning and problem solving, the process of converting verbal rules into mental procedures, and two hypotheses about the origins of systematic performance errors.

Constructing and implementing a plan is one approach to solving a problem, especially when a prestored solution method is not available. Plans for

*Douglas Danforth and Jean Aston designed the problem set, and collected and transcribed some of the protocols. James Greeno, who collaborated substantially on this research, Mario Benedicty, Friedrich Hesse, Leopold Klopfer, Sylvia Kuzmak, Jill Larkin, Pearla Nesher, Susan Omanson, and Valerie Shalin commented on written or oral drafts of this report. Special thanks to Allan Collins for his detailed review. Portions of this paper were presented at the Midwestern Psychological Association meetings, Detroit, 1981, and are available as ERIC report #ED 204 156. An earlier version of this paper appeared as an ONR Technical Report (ONR-APS-9) from the LKDC at the University of Pittsburgh, May 1982. This paper is based on a Masters thesis submitted to the University of Pittsburgh. The reading committee was James Greeno, Lauren Resnick, and Lynn Cooper. This research was sponsored by the Office of Naval Research, under Contract N00014-79-C-0215, Contract Authority Identification Number NR 667-430. Correspondence and requests for reprints should be sent to Seth Chaiklin at the Learning Research and Development Center, University of Pittsburgh, 3939 O'Hara Street, Pittsburgh, PA 15260.
problem solving can also be acquired from external sources such as another person or a book. Such plans are often stated in a brief verbal form that will be called a verbal rule. Plans described verbally are known more commonly as instructions. Verbal rules are distinguished from other kinds of instructions because they are succinct plans that apply generally to a class of problems.

There are many cognitive skills for which verbal rules have been formulated and used felicitously. A student of spelling is told, "Put i before e, except after c." This is a verbal rule because it applies generally to spelling many different words, rather than a specific word. A contemporary calculus textbook recommends to its readers that verbal rules may be a better way to learn differentiation formulas: "It is best to learn differentiation formulas . . . in words rather than with particular letters such as u and v." For example, the text’s verbal rendition for how to differentiate a product is, "The derivative of a product is the first times the derivative of the second, plus the second times the derivative of the first." (Fraleigh, 1980, p. 71). A verbal rule from signed-arithmetic is: When adding unlike-signed numbers, take the difference and keep the sign of the larger. This rule will be discussed further in the section that reports an empirical study.

Apparently the psychological properties of verbal rules and their role in problem solving have not been analyzed specifically, though other investigators have proposed ideas similar to the verbal rule and noted its potential importance for instruction (Beilin, 1976; Gagne, 1966; Gagne & Briggs, 1979; Petitto, 1978; Snyder, 1971). This paper, in three main sections, attempts to develop a general understanding of verbal rules and their role in problem solving. First, a theoretical analysis articulates properties of verbal rules and implications for using verbal rules to solve problems. Then an empirical study is reported that examines characteristics of problem-solving performance when verbal rules are instructed as a problem-solving method. Finally, the general role of verbal rules in problem solving and some implications for learning procedural skills are discussed.

THE NATURE OF THE VERBAL RULE

What is a Verbal Rule?

Besides being linguistic, verbal rules have two defining properties. First, as noted before, verbal rules describe plans for performance that can be applied to a class of problems. Second, they are usually stated in a concise, imperative form.
Why are Verbal Rules Useful?

Usually procedural knowledge can be communicated only by demonstrating its application in specific cases. The concise, linguistic form of verbal rules permits the communication of general procedural knowledge in a relatively easy manner; the linguistic form may also be easier to memorize and retrieve than sequences of actions. Thus, verbal rules may be a useful instructional method for procedural tasks. For example, that the rule “i before e, except after c” is easy to communicate, memorize and retrieve is probably verified by the reader’s experience with this rule.

What Kind of Plan is Described?

The conciseness of verbal rules corresponds to the kind of plan contained in verbal rules. Typically, verbal rules tend to be high-level plans: the verbal descriptions of the problem-solving actions are specified abstractly relative to the actual procedures that must be performed. Verbal rules may also describe application conditions stated with some degree of specificity, but usually this description is minimal. For example, the spelling rule does not specify when or where it should be used, nor does it articulate that “i before e” refers to the order in which the letters are written, nor the specific conditions of the exception clause. The calculus rule simply notes that it is applied to “products,” without discussing how to identify the factors, for example.

Consequently, like other abstract plans, plans described by verbal rules must be elaborated into specific procedures that enable successful application of the plan. This may involve supplying or supplementing the application conditions, and further specifying the action descriptions. The context in which a rule is given may provide adequate information from which a person could infer application conditions for when to apply that rule. In the signed-arithmetic rule, the first part is an explicit application condition. The condition is necessary because there is more than one verbal rule in this domain of arithmetic. A person may add more specific descriptions of what conditions constitute “adding” and “unlike-signed numbers.” The action descriptions of this rule may be further specified also. The intended interpretation of the instruction, “Take the difference,” is that the absolute value of the numbers be used.

Converting Verbal Rules into Performable Procedures

A verbal rule must be interpreted if the plan it describes is to be performed. This process involves interpreting semantically the objects referred to in the
rule, and interpreting the action descriptions by organizing appropriate procedural knowledge to realize these actions as problem-solving operators. The interpretation of a verbal rule is similar to the process of understanding written problem instructions (Hayes & Simon, 1974; Simon & Hayes, 1976), where a representation of problem states and a set of operators for changing them are constructed.

The identification and interpretation of action descriptions might be facilitated by a syntactic property of some verbal rules. The conditions for applying the plan described by a verbal rule are sometimes distinguished from the actions by an adverbial conjunction, such as when, if, and unless. In the signed-arithmetic rule, “when” indicates the condition under which the action is applied. This indication might also contribute to the interpretation of “adding” as a condition to be satisfied and not as an action to perform.

In addition to this interpretation, three additional kinds of knowledge are needed to make performable problem-solving procedures. The first is parsing knowledge that detects the objects referred to in the rule. By way of example, in the signed-arithmetic verbal rule, parsing procedures are needed to identify reliably the signs of the numbers and the operator in the arithmetic problem. The second kind of knowledge is rule-selection procedures that determine whether features, identified by parsing procedures, satisfy the application conditions. The third is executive control procedures to organize and direct performance. In executing the plan of the signed-arithmetic rule, a person must know that the application conditions are checked first and then the associated sequence of actions is performed in a certain order if these conditions are satisfied. Again, adverbial conjunctions might be a useful guide as might another syntactic property. The linear order in which a verbal rule is stated indicates implicitly a temporal sequence for verifying its application conditions and executing its actions. In the signed-arithmetic rule, a person must first check for adding unlike signs, then take the difference, then give the sign of the larger.

The preceding analysis shows that converting a verbal rule into a performable procedure involves adding a considerable amount of knowledge that is implicit in the rule statement. This result reinforces the general point that verbal rules are not adequate for immediate application in problem-solving operations or skilled performance (Bott, 1979; Brown, Collins, & Harris, 1978; Lewis & Mack, 1982; Neves & Anderson, 1981).

As part of the process of verifying this analysis, a computer simulation model was constructed that applied the verbal rules in an interpretative fashion. The model, written in the ACTP production system language (Anderson, 1976; Greeno, 1978), was composed of parsing processes, rule selection processes, execution processes, and a general control structure to decide what to do next. The declarative representation of the verbal rules was organized hierarchically after Sacerdoti’s (1977) planning system.
Using Verbal Rules in Problem Solving

People who learn a procedure from a verbal rule will be likely to have a verbal description associated with the procedure as well. The verbal rule may participate directly in performance of a task, especially when a person is first learning a procedure, by an interpretive process that operates on the rule. A verbal rule's relative accessibility also makes it useful for regenerating a procedure or a part of one when one cannot be retrieved directly. Similarly, the verbal rule can be used in a process that verifies whether implemented procedures accomplish the specifications of the rule.

It is possible, however, to develop procedures from verbal rules that do not use the verbal rules directly in performance (Neves & Anderson, 1981). Under these conditions, verbal-rule-based procedures would be comparable to procedures learned by example (Neves, 1981/1982) or from experience (Anzai & Simon, 1979), except for the associated verbal description of the plan.

EMPIRICAL STUDY OF THE ROLE OF VERBAL RULES IN PROBLEM SOLVING

For any instructed method of problem solving, we want to know whether it is effective and where it falls short. With verbal rules, we want to know in particular if they are adequate for communicating procedural knowledge, in what ways they are not, and when they are normally used for problem solving. Results bearing on these issues were obtained from analyzing problem-solving protocols in a task where verbal rules are used frequently: addition and subtraction of positive and negative integers.

One result shows that the signed-arithmetic rules were not always adequate for communicating procedural knowledge. Unsuccessful attempts at interpreting verbal rules can result in the acquisition of incorrect procedures, which if applied consistently can be called ill-composed. Ill-composed mental procedures can produce systematically flawed performance in a manner similar to that described by Brown and Burton's (1978) BUGGY system for elementary arithmetic. A second set of results shows that parsing knowledge is crucial for the selection and execution of signed-number verbal rules, but that they did not always provide adequate parsing guidance. The third set of results describes some conditions on when these verbal rules were used directly in performance.

Method

Task. The problem set contained 36 arithmetic problems—18 addition and 18 subtraction. In order to have a wide variety of problem types, three structural features were manipulated: position of larger absolute value (larger
first or second), sign on the first number (+, −, or no sign), and sign on the second number. All combinations were used, which resulted in 18 \( (2 \times 3 \times 3) \) possibilities. Numbers between 1 and 100 were selected randomly save for observing the relative magnitude constraint. Some examples from the problem set are: 18 + −25 and −37 − +49.

Subjects. Five adult, community college students who were enrolled in a remedial arithmetic course were interviewed. These students had been instructed, as part of their regular coursework, to solve signed-number problems by applying a set of three verbal rules. The actual statements of the rules that these students received was not known, but Table I describes the operations that the students were expected to perform. Two or three weeks prior to the interviews, the students had taken an in-class exam on this material. If their instruction had been successful, they should have been able to solve signed-number problems competently at the time of the interviews.

Procedure. The students were tested individually in a private room. They were given a typed copy of the problem set and scratch paper, which they were free to use. The students were instructed to (a) read aloud each problem before starting to solve, (b) describe what they were thinking while solving each problem and how they were solving it, and (c) use their normal solution methods. No time limit on solution attempts was imposed. No feedback was given intentionally. Some students were occasionally reminded to think aloud. After solving the problem set, students were asked to describe

<table>
<thead>
<tr>
<th>TABLE I</th>
</tr>
</thead>
<tbody>
<tr>
<td>Well Specified Description of Verbal Rules for Signed-Number Addition and Subtraction</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SAME-SIGNS ADDITION RULE</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF the operator is +</td>
</tr>
<tr>
<td>sign of first number matches sign of second number</td>
</tr>
<tr>
<td>THEN add the absolute values of the two numbers</td>
</tr>
<tr>
<td>put the common sign on the result</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UNLIKE-SIGNS ADDITION RULE</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF the operator is +</td>
</tr>
<tr>
<td>sign of the first number is different from sign of the second number</td>
</tr>
<tr>
<td>THEN take the difference between the absolute values of the two numbers</td>
</tr>
<tr>
<td>put the sign of the larger absolute value on the result</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SUBTRACTION RULE</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF the operator is −</td>
</tr>
<tr>
<td>THEN change the operator to +</td>
</tr>
<tr>
<td>change sign of second number to its opposite</td>
</tr>
<tr>
<td>use addition rules</td>
</tr>
</tbody>
</table>
any rules they might have used to solve these problems. They were also questioned informally about how they were solving particular problems, and about any difficulties they had. The entire interview was tape-recorded and transcribed.

Results

The Nature of the Procedures Acquired From Verbal Rules. To explore the effectiveness of verbal rules as an instructional method, the structure of the problem-solving procedures that students acquired were examined. First, a theoretical analysis of the procedures instructed as verbal rules is given, then an analysis of a student’s acquired procedures. Finally, a comparison between the instructed procedures and the student’s procedures illustrates that students need not acquire the structure of instructed verbal rules to create effective procedures, nor are verbal rules always sufficient for helping students to acquire procedures that implement the described plan.²

Structure of the Procedures that Implement the Instructed Verbal Rules. A signed-arithmetic procedure is defined as a set of operations that produces an answer to a signed-arithmetic problem. Because the signed-arithmetic verbal rules that describe these procedures are stated in a conditional form, these procedures and the students' acquired procedures are conceptualized and described as sets of production rules (Anderson, 1983). Sometimes a procedure is described by a single production rule, as is the case for the instructed signed-number verbal rules.

The production rules that represent the intended procedures of the three instructed verbal rules are presented descriptively in Table I. The structure of these procedures will be described in terms of the specific pairing of application conditions with actions in the production rules that define the procedure. Two levels of description—type and token—are useful.

The procedures that implement the instructed addition rules have two application conditions paired with two actions. At the type level, conditions and actions are described in terms of their categorical function. The type conditions for the instructed addition rules are the “value of an operator” and “whether the signs of the two numbers are the same or different.” For brevity, let cl represent the first condition and c2 the second.

²Although the students in this study were instructed explicitly in the use of signed-number rules in the context of a mathematics course, no specific attempt was made to control the process by which students studied and practiced the instruction. The manner and degree to which students used the instructed verbal rules in developing procedures and solving problems is generally indeterminate. Consequently, the descriptions of the acquired knowledge structures are interpreted comfortably as the outcome of instruction in the context of verbal rules; rather than the outcome of a process that constructs procedures only from an interpretation of the words of the verbal rule.
The type actions are an arithmetic operation and a sign assignment. Let \( a1 \) stand for an arithmetic operation and \( a2 \) stand for a sign assignment. Thus, the structure of the two procedures that implement the instructed addition rules can be described schematically, at the type level, as \( c1 \; c2 \rightarrow a1 \; a2 \).

The instructed subtraction rule is implemented in a single production for converting subtraction problems into addition problems. The only type condition is the "value of an operator" (i.e., \( c1 \)); the two type actions are to "change signs" (\( a3 \)) and to "apply another set of rules" (\( a4 \)). The structure of the subtraction rule can be described at the type level as \( c1 \rightarrow a3 \; a3 \; a4 \).

The token level of description specifies symbols or patterns that are instances of the condition and action types. The procedures intended by the three instructed verbal rules can be described as:

\[
\begin{align*}
&c1(+) \; c2(\text{same-signs}) \rightarrow a1(\text{add}) \; a2(\text{put common sign}) \\
&c1(+) \; c2(\text{unlike-signs}) \rightarrow a1(\text{subtract}) \; a2(\text{sign of larger}) \\
&c1(-) \rightarrow a3(\text{change operator to +}) \\
&\quad a3(\text{change sign of second number}) \\
&\quad a4(\text{apply addition rules})
\end{align*}
\]

The tokens for \( c1 \) specify the particular operator, while the tokens for \( c2 \) give the pair of signs. The tokens for \( a1 \) give the specific arithmetic operation to be performed; the tokens for \( a2 \) describe the sign to be assigned to the answer; the tokens for \( a3 \) specify the signs to be changed; and the token for \( a4 \) notes what set of rules to apply.

A control structure, which is presupposed by the structure of the procedures, is needed to apply the productions to produce answers. Answers to signed-number problems must contain a number and a sign. A feature of the procedure structure that will be important in the comparison with student procedures is that each addition rule is sufficient or self-contained: once a procedure is selected, only the associated actions need be performed to produce an answer. No other procedures are applied. The subtraction procedure is also self-contained and independent in that only one production is needed to convert a problem into a form that an addition procedure can use, and it can be applied regardless of the content of the addition procedures.

**Analysis of a Student's Problem-Solving Procedures.** The procedures of one student (S2) will be described and discussed. A comparable analysis for another student, with the same general results, can be found in Chaiklin (1983).

S2's procedures appear to be structured so that two productions are needed to produce an answer. The procedures that he followed and the protocol in which he described them are shown in Table II. In Table IIa, each
statement in S2's protocol that corresponds to a production rule is identified with a number.

Table IIb contains a set of production rules that correspond to the statements in Table IIa. Henceforth these production rules will be referred to as "S2's subprocedures." The first three subprocedures (1, 2, and 3) assign a sign to the answer. These sign-assignment rules are exactly the ones used for signed multiplication. The incorporation of these rules into his signed addition and subtraction procedures may have been one source of the ill-composition of his rules. This point is developed further in the discussion.

The last two subprocedures (4 and 5) decide whether to add or subtract the absolute values of the two numbers in the problem. Subprocedure 4 was stated by S2 as applying to cases when both numbers are positive. However, it was coded as "same signs" because he performed the same ac-

### TABLE IIA

**S2's Description of His Problem-Solving Rules**

| E: | "Tell me if there were any general rules that you were using in doing these problems." |
| S2: | "Mostly the same as I did last time (a previous interview), that |
| 1 | two positives are positive, and |
| 2 | two negatives equal a positive, and |
| 3 | a negative and a positive is a negative." |
| E: | "OK. And that is to put the sign on the number, but how do you decide whether to add or subtract the two numbers together?" |
| S2: | "Well, |
| 4 | when the two signs say plus then you add them. |
| 5 | if they are plus or minus, negative or positive, then you subtract." |

### TABLE IIB

**S2's Descriptions Converted into Production Rules**

| 1 | IF | the signs of the two numbers are both positive |
| THEN | put a plus sign on the numerical answer |
| 2 | IF | the signs of the two numbers are both negative |
| THEN | put a plus sign on the numerical answer |
| 3 | IF | the sign of one number is negative and the sign of the other number is positive |
| THEN | put a minus sign on the numerical answer |
| 4 | IF | the signs of the two numbers are the same |
| THEN | add the absolute values of the two numbers |
| 5 | IF | the sign of one number is plus and the sign of the other number is minus |
| THEN | take the difference of the absolute values of the two numbers |
tion when both signs were positive or both negative. Subprocedure 5 is applied when the signs on the two numbers are different.

**Computational adequacy of the proposed subprocedures.** These subprocedures can generate most of the answers that S2 produced. The sign assignment subprocedures produced all 36 signs that S2 produced. The arithmetic-operation subprocedures produced 31 of the 36 arithmetic operations. Of the 5 violations, 1 was a subtraction calculation error, and 4 were unexplained by these subprocedures. These violations will be discussed later as part of the evidence for the proposed structure of S2's procedures.

**Structure of S2's subprocedures.** S2's subprocedures can be described with the same type conditions and actions used to describe the structure of the instructed verbal rules. S2 uses only one type condition for selecting an arithmetic operation or a sign assignment: sameness or difference of signs (c2). At the token level for c2, S2 has tests for "both signs positive," "both signs negative," and "a positive and a negative."

For the actions, S2 used the same two type actions, al and a2, that occur in the instructed rules. His tokens for al were "add" and "subtract"; and the tokens for a2 were to "write a +" or to "write a -." The structure of S2's subprocedures can be described as:

<table>
<thead>
<tr>
<th>Type</th>
<th>Token</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. c2-a2</td>
<td>c2(both +)</td>
<td>a2(+)</td>
</tr>
<tr>
<td>2. c2-a2</td>
<td>c2(both -)</td>
<td>a2(-)</td>
</tr>
<tr>
<td>3. c2-a2</td>
<td>c2(both + or both -)</td>
<td>a2(subtract)</td>
</tr>
<tr>
<td>4. c2-al</td>
<td>c2(both + or both -)</td>
<td>a2(add)</td>
</tr>
<tr>
<td>5. c2-al</td>
<td>c2(both + or both -)</td>
<td>a2(subtract)</td>
</tr>
</tbody>
</table>

The numbering of these subprocedures corresponds to the productions in Table II. The important feature of the control structure needed to apply S2's rules is that a sign assignment is selected independently of an arithmetic operation, even though both of these actions have the same application condition. The psychological implication is that the subprocedures are stored independently from each other in memory.

**Three kinds of evidence for the proposed structure of S2's problem-solving procedures.** The first and primary kind of evidence for the structure of S2's procedures comes from this problem-solving performance. Three cases are presented that are consistent with the claim that S2's sign assignment subprocedures are applied independently of his arithmetic sub-

---

*This claim presupposes that S2 used a specific parsing of problems in the X - Y form, in which the second number is parsed as a negative. Evidence is given later to support this presumption.*
procedures. These cases include 4 of the 5 arithmetic-operation violations mentioned before.

The argument is that if S2's sign assignment and arithmetic operations were linked together into single, independent "same-signs" and "unlike-signs" procedures, then one would expect consistency in the application of his procedures to these cases, especially because each case involves two comparable problems that occurred adjacently in the problem set. In fact, it was observed that for some problems S2 applied a sign assignment subprocedure that was cued by one set of features and an arithmetic subprocedure associated with another set of features. These observations are taken as support for the claim that S2 has separate sign assignment and arithmetic subprocedures that operate independently.

The first case involves problems of the form \(+X - +Y\). If S2 were to follow his stated procedures for problems of this form, then he should apply his subprocedures for when the signs of the two numbers are both positive (1 and 4 in Table IIb); the associated actions are to add the two numbers and put a positive sign on the answer. The protocols for the specific problems in this case (and for the other two cases) are presented in Table III. For problems #27 (which was the calculation error) and #28, S2 subtracted, presumably using his "negative and positive" arithmetic subprocedure (5 in Table IIb). However, S2 seems to have used his "both positive" subprocedure for sign assignment because he put plus signs on the answers. This divergent application of his rules is taken as evidence that his subprocedures are in a \(c1-a1\) and \(c1-a2\) structure rather than \(c1-al\ a2\) structure. If he had a single, integrated "same-signs" or "both positive" rule, then according to his usual performance on such problems he should have added in #27 and #28, rather than subtracted.

The second case occurred with a pair of adjacent "unlike-signs addition" problems. On problem #15, S2 applied a sign assignment subprocedure cued by "a positive and a negative" for the sign assignment; but, inconsistent with his stated procedures, he applied his arithmetic subproce-

---

1The fifth violation was "17 - 16" to which S2 replied "plus 1". On a strict application of his procedures he should have added the two numbers, but it seems reasonable to assume that he simply applied his "ordinary" subtraction procedure here and did not use his procedures for signed arithmetic.

2One might object to using examples of performance that do not correspond to the proposed procedures on the grounds that these deviations will not reveal anything about the typically-used procedures. In contrast, the assumption made here is that the performance errors observed for these students result from deviations from their normal procedures. Therefore, errorful performance will reveal the subprocedures of the typical procedure because each subprocedure is vulnerable to mis-selection.

3X and Y refer to positive integers. This convention will be followed throughout the paper.
### TABLE III

**Protocol Evidence for the Structure of S2’s Problem-Solving Procedures**

<table>
<thead>
<tr>
<th>Sequence Number in Problem Set</th>
<th>Problem, S2’s Answer and Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Case 1</strong></td>
<td></td>
</tr>
<tr>
<td>#27</td>
<td>(+33 - +10 = +20)</td>
</tr>
<tr>
<td></td>
<td>“Positive 33 and a minus positive 10. That would be positive 20.”</td>
</tr>
<tr>
<td>#28</td>
<td>(+31 - +32 = +1)</td>
</tr>
<tr>
<td></td>
<td>“Positive 31 minus a positive 32, would be a positive 1.”</td>
</tr>
<tr>
<td><strong>Case 2</strong></td>
<td></td>
</tr>
<tr>
<td>#15</td>
<td>(-41 + +10 = -51)</td>
</tr>
<tr>
<td></td>
<td>“Negative 41 plus positive 10 would be negative 51.”</td>
</tr>
<tr>
<td>#16</td>
<td>(-15 + +34 = -19)</td>
</tr>
<tr>
<td></td>
<td>“Negative 15 plus a positive 34. That would be, that would be negative, uh, negative 19.”</td>
</tr>
<tr>
<td><strong>Case 3</strong></td>
<td></td>
</tr>
<tr>
<td>#35</td>
<td>(-44 - -26 = +70)</td>
</tr>
<tr>
<td></td>
<td>“Negative 44 minus a negative 26. That is 60. 70.”</td>
</tr>
<tr>
<td>#36</td>
<td>(-3 - -33 = +30)</td>
</tr>
<tr>
<td></td>
<td>“Negative 3 minus a negative 33. That would be a positive 30.”</td>
</tr>
</tbody>
</table>

A procedure that is cued usually by “both positive” (or “both negative”). On the very next problem, #16, he applied his “a positive and a negative” subprocedures in a manner consistent with his rules in Table II. The mixed application of his subprocedures on these two adjacent problems suggests that sign assignment and arithmetic operations are not linked in a single production rule.

In the final case, problem #36, S2 used the “both negative” sign assignment with the “a positive and a negative” arithmetic operation. A tempting explanation is that the origin of the subtraction operation was from an “atmosphere” effect created by the three ‘–’ signs in the problem. This explanation may not be very powerful because S2 applied the rules given in Table II on the immediately preceding problem, #35, of the same form.

Two additional kinds of evidence provide circumstantial support for the proposed structure of S2’s procedures. The first kind comes from a postexperimental interview in which S2 was asked to report his problem-solving rules. He described his sign assignment rules; the operation rules were reported only after a specific request (see Table IIa). While declarative and procedural knowledge need not be identical, this separation of reports is consistent with the claim that S2’s choice of arithmetic operations and sign assignment procedures are not joined in single production rules.
The other kind of evidence for the proposed description of S2's procedures is that his sign assignment subprocedures are exactly the same as the subprocedures used for sign assignments in multiplication. S2 had in-class instruction in multiplying signed numbers prior to the day of his interview.

**Comparison of the Structure of S2’s Procedures with the Structure of the Procedures Described by the Instructed Verbal Rules.** The first of three major differences is that S2 has separate subprocedures that operate independently for producing an answer, while the instructed rules have these operations integrated into single rules. The instructed addition rules have two type application conditions, while S2 has only one type condition in each of his subprocedures. Also, each instructed addition rule has two actions, while S2 apparently has only one action for each subprocedure.

S2 uses the same type condition (c2) from the instructed rules to select an arithmetic operation (a1) and a sign assignment (a2). However, the proposed structure of his subprocedures suggests that he does not compose these two actions into one procedure like $c2 - a1 a2$. Rather, he seems to have separate subprocedures composed like $c2 - a1$ and $c2 - a2$, with each subprocedure being selected and applied independently of the other. This form contrasts with the $c1 c2 - a1 a2$ structure of the instructed addition rules.

The second difference is that S2 does not have a separate procedure for subtraction, while the instructed rules do. At the type level, none of S2's subprocedures use the arithmetic operator as an application condition. Thus, he does not have a specific subtraction procedure nor separate procedures for addition and subtraction as found in the instructed rules. He applies the same rules to subtraction problems and to addition problems.

The third difference is that some of S2's token conditions and actions are different than the instructed tokens. S2 does not have a $cl$ (i.e., value of operator), so it is not possible to compare his rules and the instructed rules on that condition. For $c2$ (sameness or difference of signs), he distinguishes "both signs positive" and "both signs negative," and "a negative and a positive," while the instructed token of $c2$ is "same signs" and "unlike-signs." Because there is no $cl$, S2 does not have a subtraction rule that is comparable to the instructed procedures. He uses the same tokens for the $a1$ (arithmetic operation) as the instructed rules, but his $a2$ (sign assignment) differs from the token of the instructed version in that he simply writes a "+" or "−", depending on the signs of the numbers, while the instructed procedures use the relative magnitude of the numbers in one case and the signs of the numbers in another.

Thus, we see that S2 has acquired procedures from instruction with verbal rules that are not structured like the intended procedures. However, three additional properties about S2's acquired problem-solving procedures
(and those of the other students who were studied) help to clarify the validity and generality of this analysis. First, the student procedures usually contained the correct type and to a lesser extent the instructed token of the types in their application conditions—the signs of the two numbers and the operator. They did not, for example, include the sign of the first number alone, or the sign of the second number alone. This may be a consequence of the small number of possible features to use, but it could also be a result of using the verbal rules as a guide to which features should be incorporated into their procedures. Second, students were able to execute correctly the component cognitive skills (e.g., adding, determining the sign of the larger number) that implement their procedures. S2 could readily execute the components of his procedures, which accounted adequately for his answers. This seems like a weak point, but it is an important one because it supports the idea that the cause of the incorrect performance lies in the structure of a student's procedures, and not in an inability to correctly perform the component actions described in the instructed rules. This ability to perform the component skills correctly is consistent with Brown and Burton's (1978) observation for multi-digit subtraction performance. Finally, S2's ill-composed procedures are not isolated instances. Birenbaum (1981) identified these ill-composed procedures, among others, in the signed-number problem solving of 127 middle-school children who were taught similar verbal rules for solving signed-number problems.

Role of Parsing Procedures in Rule Selection and Execution. Parsing knowledge was identified in the first section as necessary for selecting and executing verbal rules. Verbal rules can be a guide to the features in a problem that determine which rule to select or what symbols to operate on; however, they may not provide sufficient guidance for how to identify those features when parsing or how to evaluate that it was done properly. This shortcoming shows an important limit on the completeness of signed-arithmetic verbal rules as a plan for performance. Three results show that students may know, in general, how to apply procedures based on verbal rules (either instructed ones or their own), but that occasional parsing difficulties can lead to mis-selecting or misapplying procedures.

Effect of Parsing Procedures on Selection of Problem-Solving Procedures. The common method for selecting a prestored procedure is to match features in a problem identified by parsing procedures against the procedure's application conditions. The particular rule selected depends on a person's interpretation of the features in a problem.

S2's rule selection depends only on the signs of the two numbers in the problem. For the problems discussed here, he used both his "ordinary" subtraction procedure and his signed-number rules (see Table II). On the
first subtraction problem of the interview, $17 - 16$, S2 apparently used his "ordinary" subtraction procedures; rather than his signed-number procedures. He said "Positive 17 minus positive 16. That would be plus 1" (these and subsequent italics added). However, on the problem, $+46 - 12$, he parsed the second number as negative rather than positive, saying, "Positive 46 and a negative 12. (pause) That would be negative 34." He has applied his "unlike-sign" rules here (3 and 5 in Table IIb). He did the same thing on $+25 - 42$ to produce $-17$. S2 also applied these rules on the 4 problems of the $X - Y$ form, in each case reading the $Y$ as a negative in the protocol.

Given that S2 parsed the sign of second number in "17 - 16" as positive, he presumably could have parsed the "12" as positive in "$+46 - 12$" as well, solving it with his "ordinary" subtraction procedures, or with his signed-number procedures for both positive numbers (1 and 4 in Table IIb). However, his parsing of the "12" as negative led him to select his signed number procedures for "unlike signs." Alternatively, this parsing may have followed his decision to use signed-number rules; rather than leading to their selection. This general point is taken up in the next section.

Effect of Problem Classification on Procedure Selection. Although in most cases a procedure associated with a verbal rule is probably selected by matching the outcome of a parse with the procedure's application conditions, there may be another selection process as well. Verbal rules can have a label that describes the class of problems to which they apply. Apparently a more general selection process may sometimes operate in which a person decides whether a problem is in the class described by the rule's label. In using this process, a person only decides that a set of rules are appropriate; a specific rule is not selected. This decision, however, can affect whether a person will attempt to use that set of rules.

For signed-number arithmetic, students try to decide, on some occasions, if a problem is a signed-number problem. This decision affects whether the student will even try to apply signed-number rules to that problem. Comparable phenomena occur in other domains. Expert problem solvers in physics can usually identify a textbook problem as solvable by energy equations or force equations, without always knowing simultaneously which specific equation(s) should be used (Chi, Feltovich, & Glaser, 1981). In high school geometry, students classify problems as being solvable with general methods such as congruent triangles or angles formed by parallel lines before actually attempting solutions (Greeno, Magone, & Chaiklin, 1979).

Two protocols illustrate situations in which students seemed to evaluate whether signed-number rules were an appropriate general method for a particular problem. These students were able to carry out parsing opera-
tions needed for selecting a signed-number rule, but they hesitated or were reassured by a process that decided whether signed numbers were involved.

S1 selected and applied the instructed rules correctly. Yet, she was uncertain about the appropriateness of using these rules for “6 − 10” because she was uncertain whether “6” and “10” were signed numbers. She first deduced the correct answer using her knowledge of “ordinary” subtraction, but abandoned this answer when asked to explain what she did. She was reluctant to use her subtraction rule because, “see this rule is for when you’re changing signed numbers and neither of these are for signed numbers.” Nonetheless, she retrieved and applied the rule correctly, but was still unsure whether it was appropriate for this problem.

S5 was given a simple addition, “47 + 7”, which could be solved with either “ordinary” addition knowledge or with signed-number procedures. Her protocol shows that she treated this problem differently than “ordinary” addition, because she considered and verified that it was a signed-number problem (see Table IV). Accordingly, S5 turned away from her “ordinary” addition procedures which probably would have produced a correct answer.

<table>
<thead>
<tr>
<th>S:</th>
<th>“47 plus 7. These are signed numbers, right?”</th>
</tr>
</thead>
<tbody>
<tr>
<td>E:</td>
<td>“Yes.”</td>
</tr>
<tr>
<td>S:</td>
<td>“OK. The way I was taught to do this, you have here a positive 7 and it’s an invisible plus sign next to the 7. So I would say that would be a— I don’t know if you would subtract or add. I would say that when you add two positive numbers it comes out negative.”</td>
</tr>
</tbody>
</table>

These examples show that a person may either be directed or hesitate to select rules on the basis of a problem classification. The classification in these examples relies presumably on a parsing that simply determines the presence of signed numbers. This result is another example of how verbal rules do not contain guidance for parsing problems or determining the appropriateness of their application.

The use of class descriptions in rule selection shows an aspect of knowledge not included in the discussion of converting verbal rules into performable procedures. This selection process may be an important theoretical addition to the representation of procedures constructed from verbal rules. Indeed, S2’s parsing of the “12” as negative in “ + 46 − 12” may have been a consequence of his initial decision that it was a signed-number problem. Because this rule selection uses only the signs of numbers, he may have simply took the visible signs. In contrast, if he had classified this problem as “ordinary” subtraction, he probably would have interpreted the “12” as positive, as he did with the “16” in “17 − 16”.

TABLE IV
S5’s Protocol for “47+7”

| S: | “47 plus 7. These are signed numbers, right?” |
| E: | “Yes.” |
| S: | “OK. The way I was taught to do this, you have here a positive 7 and it’s an invisible plus sign next to the 7. So I would say that would be a— I don’t know if you would subtract or add. I would say that when you add two positive numbers it comes out negative.” |
Effect of Parsing on Execution of a Selected Rule. Parsing knowledge is also important for the proper execution of procedures: it supplies the specific data required by the procedures. If these data cannot be determined properly, a person may have trouble executing a selected procedure.

Consider the beginning of Sl's protocol for "39 – –25." "39 minus negative 25. You change the sign of the number being subtracted. So you change 25 to a positive and you change—. I just hope that I'm changing the right sign." Sl, who used the instructed rules correctly, had selected a procedure to apply, but was unsure that actions specified by the rule were applied to the proper symbols. I attribute the cause of this difficulty to lack of adequate parsing knowledge. This was the first problem of the form X – –Y that Sl had seen in the interview, thus she was not sure that she was changing the sign specified in the action. It seems unlikely that she had forgotten the rule, because on the previous problem (10 – + 49) she recalled the entire subtraction rule and explicitly applied it step-by-step.

The results about parsing reinforce the idea that successful selection and execution of plans described by verbal rules depend on more than being able to recite the rules from memory. A person must also be able to accurately parse problems if he or she is to apply stored verbal rules consistently. Moreover, verbal rules, at least for signed arithmetic, seem ineffective for providing guidance for how to parse.

Conditions Under Which Verbal Rules are Used Explicitly in Performance

Effect of Practice on Use of Verbal Rules. Although students may learn to solve certain kinds of problems with verbal rules, students need not persist in retrieving and interpretatively applying these rules. With sufficient practice, cognitive skills can apparently be organized in such a way that a person can omit explicit recall and interpretation of verbal rules in problem-solving performance (e.g., Fitts, 1964; Neves & Anderson, 1981).

A number of protocols show that students compile verbal rules into efficient procedures over the course of solving a set of problems. The decreasing use of verbal report of some portion of a verbal rule over a set of problems during thinking-aloud problem solving is used as evidence that the verbal rule is not being applied interpretatively. This result is not a consequence of the students becoming familiar with the interview procedure, thereby saying less as the interview proceeded. Some explicit recalls occurred even on the 25th problem if it was a new form or if such a form had not been solved recently.

On the first unlike-signs addition in the problem set, Sl explicitly recalled and used a verbal rule. Her protocol was:

OK. Thirty plus a negative 29. OK. Now what I do, the first thing I do is take the largest absolute value of these numbers which would be 30 and
subtract the negative 29 and keep the sign of the largest absolute value.
So the answer would be 1. It would be positive 1. Because the larger absolute value is positive.

On the succeeding problem, also an unlike-signs addition problem, much of the verbal description disappeared. She proceeded more quickly to the solution and mentioned only part of the rule. On the seventh unlike-signs addition in the problem set, there is no evidence that verbal rules were used in a central way as in the first instance, and a number of explicit steps were dropped from her performance. The problem was \(-35 + 39\) and her protocol was "Negative 35 plus positive 39. That's subtracting 4. That would be positive 4."

This gradual elimination of the direct use of verbal rules shows their optional role in problem solving. Competent performance of a procedure learned in the context of a verbal rule does not have to use that rule directly. This result confirms the well-established empirical and theoretical result that performance becomes abbreviated with practice (Fitts, 1964; Gal'perin, 1969; Lewis, 1978/1979; Neves & Anderson, 1981; Rumelhart & Norman, 1978), and shows that this general point holds as well for performance of procedural skills learned in the context of verbal rules.

Use of Verbal Rules for Generating Procedures. Students must sometimes solve problems they have not encountered recently. If a directly applicable mental procedure is not readily available, then there is a tendency to attempt to retrieve a verbal rule to apply interpretatively. This result is an instance of Ryle's (1946) claim that "people who appeal much to principles show they don't know how to act." It also shows that a verbal rule can be used as a guide for constructing or retrieving a procedure.

For example, S1 generally solved problems smoothly, but tended to revert to explicit verbal rule retrieval when a problem appeared that was solved by a signed-number rule she had not used yet or had not used recently. One example was already given in the previous section when S1 encountered the first instance of an unlike-signs addition problem (see her protocol for "30 + -29"). In another instance, while solving the unsigned problem, 47 + 7, she said: "47 + 7. 7 and 7 is 14, plus 1, 4 and 1 is 5—54." On the next problem in the problem set, 27 + + 8, she retrieved a complete verbal rule before actually solving the problem. Her protocol for this problem was: "27 plus a positive 8. Ok. They're both positive numbers so you just add them normally and keep the common sign. So that would be—, 7 and 8, that would be 15. And carry your 1. Positive 35." Presumably, the explicit plus sign on the second number cued S1 to use signed-number rules to solve this problem, the first instance of an explicitly-signed problem in the interview. There is no indication that she used a signed-number rule on the previous problem; so when faced with what was a new problem type, from her point of view, she retrieved a verbal rule.
This result, in conjunction with the point of the last result that explicit recall and use of verbal rules tends to disappear with practice, illustrates that verbal rules are not used directly in skilled performance. Rather, they are used in attempts to patch, strengthen, or circumvent existing mental procedures that were inadequate at a given moment.

GENERAL DISCUSSION

The introduction discussed properties of verbal rules and their consequences for the learning and performance of procedural skills. In this section empirical support for these consequences are discussed. The observed performance also provides information that bears on more general questions about the nature of procedural skills. There is discussion of the implication of these results for hypotheses about (a) the process of converting verbal statements into problem-solving procedures, and (b) the origins of systematic errors in skilled performance.

The Role of Verbal Rules in Learning Procedures and Problem Solving Performance

Signed-number verbal rules are viewed reasonably, at first glance, as straightforward, complete descriptions for solving a class of problems. It is striking that learning correctly a set of seemingly simple rules can be difficult, even for adults in a community college. However, a considerable amount of knowledge must supplement a verbal rule if a person is to solve problems with the plan contained in the rule. Incorrect procedures may be introduced in the process of adding this knowledge. This is probably the major cause of the numerous systematic errors observed in the performance of students taught verbal rules to solve signed-number problems. Given that verbal rules are not a perfect method for enabling students to create appropriate problem-solving procedures, what role might verbal rules play in learning and performance?

The empirical study has shown that verbal rules have at least five specific roles in learning how to solve, and in solving problems. First, verbal rules provide a way to communicate procedural knowledge. Verbal rules, perhaps with demonstrations of their application, identify some or all of the features needed to select and apply the procedures described by the rules, and describe the needed actions. Students may have used this information to guide their initial learning of the procedures implied by the signed-number rules because they used the correct type conditions and actions in
their procedures, even if the structure of their procedures was not always correct.

Second, competent problem solvers may not use verbal rules explicitly in their problem solving, but verbal rules can play an important role in the development of efficient mental procedures. If a procedure is not immediately available for solving a problem, then a person could attempt some general problem-solving process such as means-ends analysis, or use a verbal rule, if one were available, to aid in constructing an effective procedure.

Third, just as a verbal rule can be used to construct a new procedure, it can also provide a set of imperatives to reconstruct an action sequence if the problem solver has forgotten what to do. This use of verbal rules as a problem-solving method is particularly feasible because they can be memorized and retrieved verbally. In this case, a person must decide which recalled plan to use and possibly how to implement it, but surely this is easier than constructing and implementing a plan from scratch. This adds a new and useful point to Sacerdoti’s (1977) planning formalism—a verbal rule provides a guide to action that is stored in a relatively accessible verbal form.

The students who were interviewed tended to attempt to recall verbal rules rather than use general methods to solve problems (e.g., using the semantics of positive and negative numbers to perform some sort of counting operation). This result may also be taken as evidence that verbal rules are easy to retrieve because this seemed to be the method that students preferred first.

Fourth, verbal rules can be used in a process to verify that procedures were performed correctly. The protocols of students recalling verbal rules when they were not sure what to do illustrates that verbal rules can be used to evaluate the correctness of performance.

Fifth, a verbal rule can be used to decide whether a set of procedures is appropriate for a particular problem. A verbal rule has a label that describes the class of problems to which it applies. A person may decide to reject or apply a rule depending on whether a problem has features in the class referred to by the rule’s label.

**Constructing Problem-Solving Procedures from Verbal Rules**

Converting verbal rules into problem-solving procedures is a special case of the process of converting declarative knowledge into procedural knowledge. A person might learn a procedure from a verbal rule by interpreting and applying each instruction step by step. Neves & Anderson’s (1981) proceduralization mechanism is a plausible account of such a process. This automatic mechanism would apply verbal rules interpretatively, but would also create
a specific production rule to do another problem with the same numbers and operator. After some number of problems are solved with a particular rule, enough specific productions will have been formed so that another mechanism can create automatically a generalized form of these specific productions. This generalized knowledge would be stored in a procedural form. This practice-driven mechanism thereby transforms interpretative application of a declarative description of a procedure into smooth, compiled performance.

Students may sometimes be unable to interpret and proceduralize verbal rules directly by linguistic processes. An empirical result of this study showed, unsurprisingly, that explicit instruction with verbal rules is not always sufficient for helping students acquire the described plans. Moreover, such instruction does not prevent students from constructing procedures structured differently from the instructed rules. I hypothesize that the students' primary difficulty in learning to solve signed-arithmetic problems resides, not in learning the proper conditions and actions, but in learning the proper structure of those elements. The analysis of the student problem-solving procedures shows that their procedures have the same type components as in the instructed rules, and it seems that the students knew how to perform the tests required by the conditions and the specified actions. The differences arise in the particular tokens of the type components and in the pairings of conditions and actions.

These results suggest that additional learning mechanisms, that may or may not use verbal rules in a central way, will be needed to describe processes by which verbal rules are developed into performable procedures. I conjecture that students can develop effective procedures by using or modifying their existing procedural knowledge with analogical reasoning and overgeneralization. These two processes could lead students to acquire structurally different problem-solving procedures.

Both S2 and S3 (who is discussed in Chaiklin, 1983) may have overgeneralized, respectively, knowledge about signed multiplication and "ordinary" arithmetic in the process of constructing procedures from the instructed signed-number rules. In S2's case, he may have used the structure of signed-number multiplication and division (SMD) as a guide for composing his addition and subtraction procedures. By using this existing procedural structure, he could construct procedures for signed addition and subtraction that are structured differently from the instructed rules.

Three features of S2's problem-solving procedures for signed addition and subtraction are comparable to the procedures in SMD. In SMD, sign assignment is done independently of the arithmetic operations, and the same sign-assignment rules are used in SMD for both multiplication and division. In S2's signed addition and subtraction, he applied the sign assignment and arithmetic operation independently and used the SMD sign-as-
signment rules for both addition and subtraction problems as well. The third similarity is that no sign conversions are made before applying the procedures in SMD, and S2 made no conversion on either addition or subtraction problems.

If S2's knowledge of SMD was firmly established, then he may have used this structure in some analogical fashion (e.g., Sussman, 1975; McDermott, 1978; Bott, 1979). Or more likely in this case, since the four operations were learned in close succession, S2 may have subsumed his procedures for all signed-arithmetic into a single schema. Similar application conditions for SMD procedures and the addition and subtraction verbal rules could have cued S2 to apply the control structure of SMD to signed addition and subtraction as well. In effect, this is an overgeneralization of the SMD rules. This general idea of a learning process being the application or modification of existing procedures to new problem situations is consistent with proposals for algebra learning by Davis, Jockusch, & McKnight (1978), Matz (1980) and others.

The Nature and Origins of Systematic Errors in Skilled Performance

Procedures involved in skilled performance can have systematic flaws that produce consistent errors in the results of this performance (Brown & Burton, 1978). Procedures constructed from verbal rules are no exception.

Some proposals about the origins of systematic performance errors have claimed that faulty procedures arise from substituting incorrect actions for individual steps in an otherwise correct procedure and/or incompletely learning a procedure (Brown & VanLehn, 1980; Young & O'Shea, 1981). It is possible to make slight modifications of the correct signed-number procedures (e.g., deleting an action) to produce the same answers as the flawed student procedures. However, the analysis of the structure of student signed-number procedures yields two additional explanations for the specific origins of some faulty procedures in skilled performance.

First, it was suggested that existing procedures from related domains were being substituted or adapted for signed addition and subtraction. This use of other existing procedures, perhaps motivated by semantic considerations, is a kind of mechanism for creating faulty procedures that contrasts with the piece-meal modification of individual steps in a procedure. Larkin (1978) noted a comparable process in adult solutions of fraction problems.

A second mechanism that could create procedures that yield systematic errors in performance is a process that proceduralizes remembered rule fragments. S3 added a sign conversion to his problem-solving procedures on the basis of his memory for his teacher's description. There may have been no impasse or incomplete procedure here to be "repaired" (Brown & VanLehn,
1980), rather a procedure was added because of rational considerations. Alternatively, there may have been an impasse and this construction reflects a general problem-solving method used for repair: namely, recall and proceduralize an instructed description of how to solve the problem.

CONCLUSION

This paper has tried to clarify the nature of verbal rules and their role in problem solving. It seems likely that verbal rules have been developed as a general problem-solving method because they provide an efficient method for coding a plan into a concise and easily retrievable form. The analysis in this paper has identified properties of verbal rules that enable performance. The empirical study has revealed limiting conditions on the effectiveness of verbal rules in problem solving and problems that can arise when persons try to use them.

The main points of these analyses are now stated in summary. Students must add a non-trivial amount of knowledge to make the rules effective. Adequate parsing knowledge is needed if the rules are to be effective. Providing verbal rules as a technique for solving problems does not guarantee that those rules will be used in a central way, nor that they will be learned the way they were taught. Moreover, students can acquire procedures that are different from the structure of instructed rules. Flawed procedures can arise in attempting to elaborate the verbal rules into performable procedures. A hypothesized source of flaws is the use and possible overgeneralization of other knowledge for constructing procedures. A second source is an incorrect memory of an instructed verbal rule that is incorporated into a problem-solving procedure. It was shown that, while verbal rules tend not to be used in skilled performance, they are useful for recovering from inadequate or unavailable procedures. A person can appeal to concise, accessible verbal statements as a way to problem solve rather than acquiring new procedures by constructive methods.

REFERENCES


