Representation of Similar Well-Learned Cognitive Procedures

RENÉE Elio
Carnegie-Mellon University

Continued practice on a task is characterized by several quantitative and qualitative changes in performance. The most salient is the speed-up in the time to execute the task. To account for these effects, some models of skilled performance have proposed automatic mechanisms that merge knowledge structures associated with the task into fewer, larger structures. The present study investigated how the representation of similar cognitive procedures might interact with the success of such automatic mechanisms. In five experiments, subjects learned complex, multistep mental arithmetic procedures. These procedures included two types of knowledge thought to characterize most cognitive procedures: "component" knowledge for achieving intermediate results and "integrative" knowledge for organizing and integrating intermediate results. Subjects simultaneously practiced two procedures that had either the same component steps or the same integrative structure. Practice-effect models supported a procedure-independent representation for common component steps. The availability of these common steps for use in a new procedure was also measured. Steps practiced in the context of two procedures were expected to show greater transfer to a new procedure than steps learned in the context of a single procedure. This did not always occur. A model using component/integrative knowledge distinction reconciled these results by proposing that integrative knowledge operated on all steps of the procedure: An integral part of the knowledge associated with achieving an intermediate result or state includes how it contributes to later task demands. These results are discussed in the context of automatic mechanisms for skill acquisition.

The phenomenon that people get faster at a task with continued practice is ubiquitous. In fact, the speed with which a person executes a pro-

This research was conducted while the author was at Carnegie-Mellon University and supported in part by an NSF graduate fellowship and by ONR grant N00014-81-C-0335 and NSF grant IST 80-15357 to John Anderson. John Anderson, Bill Chase, and Herbert Simon made insightful suggestions during all phases of this research. I am grateful to Peter Dixon for his extensive and valuable input on earlier versions of this manuscript. I also thank Donald Norman and two anonymous reviewers for their constructive criticisms.

Correspondence and requests for reprints should be sent to the author who is now at the Department of Computing Science, University of Alberta, Edmonton, Alberta T6G 2H1.
procedure is often a criterion for evaluating their skill level. In addition to this quantitative change in performance, continued practice brings about qualitative changes as well. On the positive side, the skill becomes more automatic. One good characteristic of automaticity is that the skill suffers less interference from other ongoing activities (Spelke, Hirst, & Neisser, 1976). On the negative side, extensive practice on a task can sometimes lead to performance that is brittle and routinized. The most well-known example of this, the Einstellung effect, comes from the problem-solving domain (Luchins, 1942). Luchins taught subjects two methods for solving a problem, a simple method and a complex method. Subjects who received extensive practice on problems solvable only with the complex method did not apply the simple method when a problem could be solved using either approach. Other subjects who had not received extensive practice on the complex method had no difficulty in recognizing when the simple method could be used.

Another negative consequence of extensive practice is that knowledge associated with well-practiced procedures is sometimes less “accessible” for application in new situations. Often, this phenomenon has been studied and observed in transfer-of-training studies. In the standard transfer paradigm, subjects practice one task and then move to a second task. For some subjects, the second task will have a feature that is similar to the first task. For other subjects, the second task will have no similarity to the first. When the similar-tasks group performs better on the second task than the dissimilar-tasks group, this result is attributed to “positive transfer” of knowledge associated with the first task to the second. Interestingly, there is evidence that, with some tasks, the degree of positive transfer decreases as the amount of practice on the first task increases (Hayes-Roth, 1977; Mandler, 1954).

A model that can account for the quantitative practice effects, such as speed-up, should also be able to account for the qualitative changes in performance as well. Newell and Rosenbloom (1981) have offered a “chunking model,” which proposes that performance on a complex task depends on the use of knowledge chunks that correspond to the task’s components. The time to perform the task is related to the number of chunks accessed. According to this model, continued practice on a task leads to the formation of larger knowledge chunks from smaller knowledge chunks.

As an example, consider how a person might learn to use a computerized forms management system. A forms management system typically allows a person to create on-line versions of paper forms, by specifying the lay-out and labelling of fields that will be filled with different types of information. A person can view a form on a CRT display, using different key presses to move among fields or to initiate a change to a field. There can be additional procedures associated with these operations. For example, if the person tries to access a field tagged “confidential,” the system might prompt for a password to permitting the access.
The general knowledge associated with using such a system might be described as consisting initially of several independent pieces of knowledge: how to view a form, how to move around the form in any direction, how to initiate field modifications (e.g., fill in, erase, change), how to make the modification, how to save the modification, and how to remove the form from the screen. To do a specific task, the relevant knowledge must be selected from everything known about the forms system, ordered in the correct sequence, and executed. Suppose a person's task involved changing the information in a confidential field for a large number of forms. For the first few forms, each step required to accomplish this task is consciously recalled and executed. As the person becomes more familiar with the task, sequentially executed steps are executed faster. For example, bringing up the form, getting to the confidential field, and initiating a modification request might be one burst of familiar keystrokes. Responding to a password prompt, typing a password, a return, and entering the actual field change becomes another burst of familiar actions. According to the chunking assumption, the initial slow, halting execution of separate steps is replaced by these smooth, rapid actions as smaller knowledge chunks are merged into larger knowledge chunks.

By itself, this chunking assumption cannot account for the speed-up associated with practice effects. It must be coupled with the assumption that the time to perform a task is related to the number of independent chunks, but not to the size of the knowledge chunks. In other words, a few large chunks will be accessed or used faster than several smaller chunks, even though the larger chunks are more complex. These two assumptions together can account for the speed-up due to practice.

One further assumption is needed to cover the qualitative changes, such as the Einstellung effect or the decreased availability of knowledge for new situations. The previously smaller, independent knowledge chunks lose their autonomy when they are merged into more complex knowledge structures. In this way, if they cannot be accessed independently of the larger knowledge chunks, they cannot contribute to a similar task or skill. The gist of these three assumptions taken together is that qualitative changes in knowledge representation underlie the quantitative changes in performance found with extensive practice. This general notion is common to several models of skill acquisition and practice effects (Fitts & Posner, 1967; Hayes-Roth, 1977; Lewis, 1978; Mandler, 1954, 1962; Neves & Anderson, 1981).

Lewis (1978) formalized these general assumptions in a computer simulation model aimed at accounting for the speed-up that characterizes extended practice on a task. He used a production system formalism for representing knowledge. In a production system, knowledge is represented as a set of IF-THEN rules, or productions. The IF part of a rule specifies a set of conditions that must be true for the rule to be used. These conditions usually reflect some particular state of the system's working memory, its
repository for information about the world or the task at hand. The THEN part of a rule specifies a set of actions that will be taken if the conditions are satisfied. Actions typically cause some change to the system's working memory, which in turn causes the conditions of some other rule to become satisfied. As a formalism for modeling human cognition and building intelligent systems, production systems have undergone considerable development (e.g., Anderson, Kline, & Beasley, 1980; Anderson, 1983; Davis, Buchanan, & Shortliffe, 1977; Forgy, 1982; Newell, 1973).

To account for both the quantitative and qualitative performance changes that occur as a function of practice, Lewis proposed a “composition” mechanism. The composition mechanism would merge sequentially executed rules into a single, large composite rule. The composite rule would specify all the conditions required to trigger all the actions previously associated with the independent rules. For example, consider the following productions for the example task of accessing confidential fields using the forms management system described earlier:

Rule 1: IF you are at a field that must be modified & the field is labelled "confidential" & the modification request is "M"
THEN hit "M"

Rule 2: IF "M" has been hit & you know the password
THEN type the password & hit return

Rule 3: IF the password has been entered
THEN type the modification & hit return

These rules correspond to small, independent pieces of knowledge associated with this particular task. A larger knowledge chunk formed from Rules 1 and 2 would be the composite rule:

Rule 4: IF you are at a field that must be modified & that field is labelled "confidential" & the modification request is "M" & you know the password
THEN hit " M"
& type the password
& hit return

Rule 4 includes all the conditions of Rule 1 plus the conditions of Rule 2. There is one Rule 2 condition, "M" has been hit, that is omitted from Rule 4. It originally appeared as a Rule 1 action. If it were included, Rule 4's IF side would be satisfied only if Rule 1 had fired. In general, conditions of one rule that depend on actions of another rule are dropped from the IF side of their composite (Lewis, 1978; Neves & Anderson, 1981). The composite of Rule 4 with Rule 3 would yield the following production:
Rule 5: IF you are at a field that must be modified & that field is labelled "confidential" & the modification request is "M" & you know the password THEN hit "M" & type the password & hit return & type the modification & hit return

In line with the general assumptions outlined earlier, the composition approach considers the time to perform a task to be proportional to the number of independent productions, but not to the size of the productions. Additionally, a composite production's conditions must be matched in an all-or-none manner. As a consequence, the knowledge represented by previously independent rules is accessible only when the more complex configuration of composite conditions is satisfied. A new situation that uses some of the knowledge may not satisfy the larger set of conditions in the composite rules. Suppose, for example, that the field label "confidential" was replaced on the computerized forms with several new specific labels (like "salary"). The same procedure for accessing these fields might still apply. However, the conditions of composite Rules 4 and 5 would not be satisfied, even though they represent some knowledge that is still relevant to the task.

Lewis' (1978) simulation did a good job in accounting for both the speed-up and the Einstellung-like inflexibility that characterizes some skilled performance. However, he demonstrated that composition would yield an exponential speed-up in performance over time, considerably faster than the power-law speed-up observed across a wide variety of cognitive tasks (Newell & Rosenbloom, 1981). Either the basic assumptions underlying an automatic composition mechanism are wrong, or there are other features of procedural knowledge and skilled performance that influence how it operates.

One important feature of any skill is how similar it is to another skill. There can be many levels and types of similarity between two cognitive skills. At a very general level, they can have similar strategic or organizational features. For example, the skills for using two different forms management systems are likely to share knowledge such as, “If I want to change a field on a form, then I must bring that form to the screen.” At the other end of the scale, two cognitive procedures might both use quite specific, identical pieces of knowledge. Thus, the knowledge associated with accessing a confidential field, used within the larger procedure for modifying forms, may also be part of other complex procedures, like defining a new form that includes confidential fields. The theoretically interesting issue concerns how similarity is reflected in the representation of two skills and whether it impacts a mechanism like composition.
This study investigated the second type of similarity between two cognitive procedures. Subjects simultaneously learned two procedures that shared some identical parts. Given the forms management example, this would be analogous to training someone simultaneously on a modifying-fields procedure and a defining-new-forms procedure. Although each procedure has some unique features, both procedures might include the identical miniprocedure of accessing a confidential field. The question is whether two procedures with some identical parts each have their own representation of the common parts, or whether they access a single, procedure-independent representation of common knowledge.

The study measured both a quantitative and a qualitative aspect of performance thought to reflect the character of the knowledge representation. The quantitative performance feature was how rapidly people execute parts common to two procedures as a function of practice. Assume that each procedure is practiced the same amount. As noted earlier, the speed-up in performance across practice trials usually follows some form of a power-law function. There are two ways to apply this to modeling the time to execute parts that are present in two procedures. The first way would count the total practice opportunities that occurred across both procedures. The second way would only count the practice opportunities in one procedure. This would be appropriate if each procedure had its own copy of the common steps. Each copy would only get as much practice as each procedure did. If changes in how fast people execute common parts over time are best modeled by the first way, this would suggest that the procedures accessed a single, independent representation of the common parts. This would also support the hypothesis that the experience of practicing some portion of a skill in two different contexts (i.e., procedures) thwarted a composition mechanism. This follows from the characterization of composition, because each procedure would present a different set of conditions for accessing the common parts. Comparing alternative models of practice effects for similar procedures will reflect a general aspect of their knowledge representation and provide one piece of evidence about how a composition-like mechanism works as similar cognitive skills are being acquired.

The qualitative performance feature this study investigated was the availability of knowledge for transfer to new procedures. If the practice models support a procedure-independent representation of identical parts, then these parts should easily transfer to new procedures. This was evaluated by comparing the transfer of parts initially learned in the context of two procedures to the transfer of parts initially learned in the context of one procedure. There should be greater transfer in the two-procedure case (if the parts have a procedure-independent representation) than in the one-procedure case. In the one-procedure case, composition should work to subsume the parts into large, procedure-specific knowledge structures. Other
studies have demonstrated that variable learning contexts benefit transfer in situations as diverse as perceptual motor tasks (Duncan, 1960) and story recall (Thorndyke & Hayes-Roth, 1979). In this way, the patterns of transfer will be a second source of evidence about how composition operated when similar procedures were learned. They should also follow from the knowledge representation implicated by the practice-effect models.

GENERAL METHOD

Component versus Integrative Knowledge

A cognitive procedure often consists of a series of discrete steps, each yielding an intermediate state or result. The procedure must specify two things: how to achieve each state and how to integrate intermediate states. Skilled performance at a task relies on both these types of knowledge. Consider a "first learn the parts, then put them together" approach to acquiring a new skill. If putting the parts together is an integral aspect of skilled performance, then the skill level achieved on each part in isolation may not carry over completely when the parts are attempted in sequence (Fitts & Posner, 1967). Conversely, when a person practices both the discrete parts and their integration simultaneously, then skilled performance on the parts may be dependent on the particular framework of integration in which they were learned. This is a restatement of the composition predictions for transfer: Well-learned procedures practiced within one framework cannot be easily lifted and used in another framework (at least, initially not at their previous level of performance). More precisely, the benefit of prior practice on the component parts of a procedure may be offset by the disruption of executing them in a new integrative scheme.

A component/integrative distinction has been used in models of reaction time tasks (Logan, 1978, 1979) and seems an important feature of cognitive procedures as well. Therefore, the procedures used here involved localized routines for obtaining intermediate results (component knowledge) and a structure for organizing and combining intermediate results (integrative knowledge). This provided an interesting context in which to ask the questions outlined earlier: Do procedures that share the same integrative structure access the same representation of that knowledge? Is there a single representation of component steps that are common to more than one procedure? Do changes in one type of knowledge impact the transfer of the other type of knowledge? The methodologies outlined earlier for investigating the knowledge representation of similar procedures were applied to both the component and integrative aspects of the procedures.
Task

The present study used specially designed mental arithmetic procedures, so that the similarity of both component and integrative features could be carefully controlled. The procedures specified how to compute a set of intermediate results using information from an external display and how to combine the intermediate results to obtain a final solution. The procedures were presented to subjects as methods for computing a quality rating for a water sample, using a set of chemical analyses. The results of these chemical analyses appeared on a CRT display. For each water sample, the display presented its "solid count," its "algae count," four measures of its "lime concentration" and four measures of its "toxin rating." An example display is shown in Figure 1.

Each procedure had six steps. The column of numbers in the lower half of the screen corresponded to these steps. A subject entered the current step's answer next to the number pointed to by the arrow. For reasons explained below, the answer was echoed only briefly, then erased from the CRT. The arrow then moved down to the next number in the column to indicate the current step.

An example procedure is given in the top panel of Table 1. Three of the procedure's six steps—calculations for Particulate Rating, Mineral Rating, and Marine Hazard—involves finding and combining numerical information from the screen. These calculations used a mix of the four standard arithmetic operations, plus judgments of greater-than, less-than, minimum,
TABLE I
Examples of Procedures for Each Transfer Condition in Experiment 1

<table>
<thead>
<tr>
<th>Procedure A: Initially Learned Procedure</th>
<th>Procedure B: Transferred Component Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Particulate Rating</td>
<td>Solid × ((lime₃) − lime₄)</td>
</tr>
<tr>
<td>Mineral Rating</td>
<td>Greater of (algae/2) (solid/3)</td>
</tr>
<tr>
<td>Index 1</td>
<td>Particulate + Mineral</td>
</tr>
<tr>
<td>Marine Hazard</td>
<td>(toxin max + toxin min)/2</td>
</tr>
<tr>
<td>Index 2</td>
<td>Index 1/Marine</td>
</tr>
<tr>
<td>Overall Quality</td>
<td>Index 2 − Mineral</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Procedure C: Transferred Integrative Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Particulate Rating</td>
</tr>
<tr>
<td>Mineral Rating</td>
</tr>
<tr>
<td>Index 1</td>
</tr>
<tr>
<td>Marine Hazard</td>
</tr>
<tr>
<td>Index 2</td>
</tr>
<tr>
<td>Overall Quality</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Procedure D: Control Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Particulate Rating</td>
</tr>
<tr>
<td>Mineral Rating</td>
</tr>
<tr>
<td>Marine Hazard</td>
</tr>
<tr>
<td>Index 1</td>
</tr>
<tr>
<td>Index 2</td>
</tr>
<tr>
<td>Overall Quality</td>
</tr>
</tbody>
</table>

and maximum. The first calculation in Table I’s Procedure A uses a subscript notation (lime₄) to indicate which of the four lime concentration values should be used. In this case, the second lime concentration reading (5) must be subtracted from the fourth lime concentration reading (9). This result must be multiplied by the solid count (6). The calculation of Mineral Rating requires computing two divisions and selecting the greater result. The Marine Hazard calculation uses the subscript notation to specify that the maximum and minimum toxin values (8 and 2, respectively) should be added and then divided by two. These three steps were termed the “component steps” of the procedure. The other three steps—calculations of “Index 1,” “Index 2,” and the “Overall Quality”—combine the partial results calculated on preceding steps to derive the final answer. The calculations for Index 1, Index 2, and Overall Quality were termed “integrative steps.”

The procedure’s integrative structure consists of more than just the three integrative calculations. Many aspects of the procedure as well as
general task demands can contribute to an integrative framework for executing a complex procedure. The integrative structure defined here included those task characteristics that would influence successful performance and hence seemed likely to impact the representation of the procedure. These task characteristics were (a) the unavailability of previously computed intermediate results on the display screen, (b) the relative ordering of component and integrative calculations, and (c) the times during the procedure when past intermediate results must be used again.

Keeping track of intermediate states and results was considered an important part of a cognitive procedure's integrative structure. This was why answers were not kept on the CRT screen. Within the context of this general task feature, the last two integrative characteristics mentioned above defined a procedure's unique memory load demands. As Procedure A in Table I illustrates, some partial results (e.g., Particulate Rating) can be forgotten as soon as they are combined into an intermediate result. Other partial results must be remembered for several steps. For example, Mineral Rating is computed on the second step but must be remembered for both the third and the sixth calculations. The availability of partial results needed for integrative steps will depend on the success of the processes devoted to keeping track of them. Although much of what is defined as integrative structure was not directly measured on integrative steps, the observable benefit of transferring a familiar integrative structure to a new procedure may be most evident on those steps.

OVERVIEW OF EXPERIMENTS

In all the experiments reported here, reaction time was the primary performance measure. Transfer of prior knowledge to a new procedure was measured as the "savings" in the time to execute a step. Savings was evaluated against a situation in which the old and new procedures were unrelated. When the old procedure is not related to the new one, there is no specific knowledge to transfer from one to the other.

The goal of Experiment 1 was to establish how component and integrative knowledge transferred from one procedure to another procedure. Subjects first learned and practiced one mental arithmetic procedure. They then learned a second procedure that either had the same component steps, the same integrative structure, or completely different steps.

Experiments 2–5 investigated whether knowledge practiced in two different training procedures showed a higher level of transfer to a third procedure. In these experiments, subjects simultaneously learned and practiced two training procedures. These procedures shared one type of knowledge (e.g., the integrative structure), but differed on the other type (e.g., the
component steps). Thus, they offered an opportunity to practice the same parts in two different contexts. The transfer of these parts was contrasted with the transfer of parts practiced in Experiment 1's single-procedure context.

EXPERIMENT 1

Method

Design. There were three experimental conditions defined by the specific knowledge that could be transferred from the first procedure to the second procedure. The second procedure could have (a) the same component steps but a different integrative structure (transferred-component condition), (b) the same integrative structure but different component steps (the transferred-integrative condition), or (c) both different component steps and a different integrative structure (the control condition). These three possible relations between the first and second procedure are illustrated in Table I.

In Table I, Procedure B illustrates the transferred-component condition. It uses the same component steps as Procedure A, but with a different integrative structure. The new integrative structure differs from the old integrative structure in three ways: the relative ordering of component and integrative steps is different; the sequence of component steps is different; and different combinations of intermediate results are combined at different times during the procedure.

The transferred-integrative condition is illustrated by the relation between Procedures A and C in Table I. In this case, Procedure C uses the integrative structure from the Procedure A: partial results are combined with the same operators, in the same combinations, and at the same times during the procedure. However, different calculations are used to compute the three component step results.

The control condition is illustrated by the relation of Procedures A and D. Procedure D differs from Procedure A in both its component steps and its integrative structure.

Stimulus Construction. One general feature of the task was considered in designing the mental arithmetic procedures. Each subject would need at least six different procedures, two for each transfer condition. To ensure nominal transfer in the control condition, the training procedure and the transfer procedure should be as different as possible. However, in the transfer conditions, the transfer procedure must have either the same component steps or the same integrative structure as the training procedure. It was easiest to satisfy both goals by first making the procedures as different
as possible. This would meet the constraints of the control condition. For the transfer conditions, the transfer procedure's component or integrative features were replaced with the corresponding features from the training procedure.

The general method for constructing these procedures was to devise six different procedure "templates." Each template specified three global features of a procedure: the relative ordering of integrative and component steps; which partial results were combined on each integrative step; and which calculations were used to compute component step results. The six templates were divided into three pairs. These pairs were fixed across all subjects.

Two of the template features determined the integrative structure: the relative ordering of component and integrative steps, and the assignment of partial results as arguments for integrative steps. The two possible orderings of component and integrative steps were component - component - integrative - component - integrative - integrative and component - component - integrative - integrative - integrative - integrative. The first ordering was called mixed ordering; the second was called blocked ordering. Each template pair consisted of one template that used mixed-step ordering and one template that used block-step ordering. The assignment of partial results as arguments to integrative steps had a large influence on the similarity of the templates. This was done in a manner that maximized template differences, first within a template pair and then across the template pairs.

Specifying the component step calculations for each template offered the widest range of variability. Three different types of component calculations were designed. The first type involved operands available from the display (e.g., solid × lime). The second type involved a magnitude judgment to obtain the one of the operands. For example, the calculation "solid + lime max" meant "add the value of solid to the maximum of the four lime values." The third type involved a magnitude judgment of the results of two calculations (e.g., lesser-of [solid + algae] and [lime i + lime j]). A pool of possible component steps was generated from these three types. One component step of each type appeared in each template. Any particular component step from the pool occurred only once across a subject's six procedures.

This method yielded three pairs of procedure templates. For each subject, the templates were filled with randomly selected operators and arguments to create six complete, unique procedures. The assignment of procedure pairs to the transfer conditions was also randomly determined for each subject. Each procedure within a pair served as the training procedure half the time. For a procedure pair assigned to the control condition, the procedure that a subject would learn second was left unchanged. In the transferred-integrative condition, the second procedure's integrative structure and steps were replaced with the first procedure's integrative structure
and steps. In the transferred-component condition, the second procedure's component steps were replaced with the component steps used in the first procedure. In addition, the order of transferred component steps was different in the second procedure than it had been in the first procedure.

Apparatus and Procedure. The experiment was controlled by a PDP 11/34 computer. The problem displays were presented on a CRT screen and subjects entered their answers using the number pad on the terminal keyboard.

At the start of each session, a subject received a hardcopy of the training procedure and memorized it until he or she could reproduce it correctly for the experimenter. Subjects were allowed to use pencil and paper to help them practice and memorize the procedure. The hardcopy was removed when they began solving problems.

Subjects solved 50 problems using the training procedure. Each problem consisted of a display of numbers like that given in Figure 1, with an arrow pointing to the current step. The numbers were randomly generated for each problem, but the program rejected a set of numbers if any of the six calculations produced a result greater than 50. Subjects used the following rules for calculating and entering their answers: (a) take the absolute value of all the answers, (b) use zero as the answer to division by zero, (c) round down fractional answers, and (d) enter only the last digit of a two-digit answer but use the whole number in subsequent calculations. (This last rule was needed because of system difficulty in accepting and timing a rapid succession of two-character responses under certain load conditions.) Each answer was erased as soon as it was entered and the arrow moved to the next step. Response time and accuracy were recorded for each of the six answers.

Subjects were instructed to solve each problem as fast as possible without sacrificing accuracy. After a subject entered the last answer for a problem, the screen erased. The total solution time and the names of any incorrect steps were then displayed on the screen. If a subject made three consecutive errors on the same step, the whole procedure was presented on the screen to review. The subject terminated this review time when he or she was ready to continue. Opportunities for rest breaks occurred after every 10 problems.

After finishing 50 problems using the training procedure, the subject was given a hardcopy of a second procedure to learn. The second procedure either had the same integrative structure, the same component steps, or was a completely new procedure. After the subject correctly reproduced the second procedure, the hardcopy was removed. Subjects solved 50 problems with the second procedure.

Subjects. Twenty-seven students from the Carnegie-Mellon University community participated in the experiment. Because all subjects experienced
all three conditions, the experiment was divided into three 2-hour sessions, which took place on 3 different days. Each of the 27 subjects had a different assignment of conditions and procedures to sessions.

Results

Performance was averaged over sets of five problems for each subject, yielding 10 scores for the first procedure and 10 for the second, transfer procedure. Both accuracy and reaction time were measured. Accuracy was not significantly affected by the various transfer manipulations. Subjects averaged between 85% and 97% on both component and integrative steps. Not surprisingly, subjects became considerably faster at executing the procedures as a function of practice. On the first procedure, subjects averaged about 10 s per step initially and about 2.5 s per step after 50 trials. These practice effects are analyzed in a later section. The following analyses concentrate on reaction time, evaluating the degree of transfer from the first to the second procedure.

**Integrative and Component Knowledge Transfer.** If transfer occurred, then there should be some savings in the time to execute old, well-practiced steps. The savings associated with transferring prior knowledge was measured as the difference between the mean reaction time per step in the control condition and the mean reaction time per step in each of the transfer conditions. Table II presents both mean times per step and these differences.

Separate analyses were done for the first 25 problems and for the last 25 problems on the second procedure. This early-versus-late division would

<table>
<thead>
<tr>
<th>Step Type</th>
<th>Component</th>
<th>Integrative</th>
<th>CTD a</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>First 25 Transfer Problems</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Control: new integrative/new component</td>
<td>5.89</td>
<td>4.16</td>
<td></td>
</tr>
<tr>
<td>Transferred integrative/new component</td>
<td>5.41</td>
<td>.48</td>
<td>2.53</td>
</tr>
<tr>
<td>Transferred component/new integrative</td>
<td>4.88</td>
<td>1.01</td>
<td>3.69</td>
</tr>
<tr>
<td><strong>Last 25 Transfer Problems</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Control</td>
<td>3.70</td>
<td>2.05</td>
<td></td>
</tr>
<tr>
<td>Transferred integrative/new component</td>
<td>3.09</td>
<td>.61</td>
<td>1.66</td>
</tr>
<tr>
<td>Transferred component/new integrative</td>
<td>3.48</td>
<td>.22</td>
<td>2.14</td>
</tr>
</tbody>
</table>

CTD = Control – transfer difference
reveal how transfer effects changed as subjects become more skilled on the second procedure. For the first 25 problems, the differences in Table II indicate that when component or integrative knowledge was transferred, there was a significant benefit on the corresponding steps, $F(1,26) = 20.6$ and 5.1, respectively, $p's < .03$. For the first 25 problems, the savings occurred where expected, namely on the steps that were transferred from the first to the second procedure.

The nature of the savings changed in some interesting ways for the last 25 problems. As the means in Table II indicate, there was still an overall advantage for using an old integrative structure with new component steps. However, the observable benefit of using old component steps in a new integrative structure was short-lived. The disappearance of the savings for transferred component steps becomes clearer by contrasting component-step time for the first and last 25 problems. In the first 25 problems, subjects were faster on old component steps than they were on new component steps used in an old integrative structure. The reverse was true for the last 25 problems. Subjects were slower on old component steps used in a new integrative structure than they were on new component steps in an old integrative structure. This interaction (between first versus last 25 problems and transferred-integrative versus transferred-component conditions) was significant, $F(1,26) = 20.0$, $p = .001$. It is notable that the time spent on new component steps in an old integrative structure (3.09 s) was significantly less than the time on new component steps in a new integrative structure, the control condition, $F(1,26) = 4.3$, $p = .04$.

**Memory-Load Effects.** To determine whether the transfer effects were influenced by the number of intermediate results subjects had to remember, each step across the six procedures were classified according to a memory-load of either 0, 1, 2, or 3. The sixth step, which was an integrative step, may have had special status because it was the last step. Therefore, it was not included in these analyses. Table III gives the mean time per step as a function of memory-load and condition for the first 25 problems on the second procedure. There was a significant benefit for using old component steps when the memory-load was 3, $F(1,76) = 6.3$, $p < .02$. Likewise, the benefit of an old integrative structure was felt on steps with a memory-load of either 2 or 3, $F(1,26) = 4.7$ and 8.7, $p's < .01$.

These data indicate that positive transfer was greater at higher memory-loads. However, memory load was perfectly confounded with serial position in this task: memory-load 0 steps were always first, memory load 1 steps were always second, and memory-load 2 steps came in the middle of the procedure. Although memory load and serial position cannot be separated, it seems intuitively reasonable that, as the demands on working memory be-
TABLE III
Experiment 1: Mean Reaction Time Per Step as a Function of Memory Load
for the First 25 Transfer Problems, in Seconds

<table>
<thead>
<tr>
<th>Condition</th>
<th>M</th>
<th>CTD</th>
<th>M</th>
<th>CTD</th>
<th>M</th>
<th>CTD</th>
<th>M</th>
<th>CTD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Component Steps</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Control</td>
<td>5.20</td>
<td>5.70</td>
<td>6.79</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Transferred integrative</td>
<td>4.65</td>
<td>.55</td>
<td>5.26</td>
<td>.44</td>
<td>6.32</td>
<td>.47</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Transferred component</td>
<td>4.78</td>
<td>.42</td>
<td>4.78</td>
<td>.92</td>
<td>5.09</td>
<td>1.70</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Integrative Steps</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Control</td>
<td></td>
<td></td>
<td>2.86</td>
<td>5.12</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Transferred integrative</td>
<td>1.89</td>
<td>.91</td>
<td>3.15</td>
<td>1.97</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Transferred component</td>
<td>2.09</td>
<td>.77</td>
<td>4.86</td>
<td>.26</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

come greater, the benefit of using much-practiced routines should increase. What these analyses show is that the distinctions between integrative and component steps cannot be attributed just to memory load (or serial position) differences. This was true for the remaining experiments as well, so their memory-load analyses will not be presented.

Discussion

There was a significant benefit for using either old component knowledge or old integrative knowledge in a new procedure. However, the transfer effects were not independent and changed as a function of practice. At the start of learning a new procedure, transferring one type of procedural knowledge did not affect learning or using the other type of procedural knowledge. This was not true by the end of practice. By that time, subjects were faster on new component steps in an old integrative structure than they were at (a) old component steps in a new integrative structure, and (b) new component steps in a new integrative structure.

It is hard to dispute that transferring a well-learned integrative structure helped performance on new component steps. Why changing the integrative structure hurt performance on old component steps has two possible explanations. The first explanation assumes that integrative processes, perhaps those devoted to rehearsing intermediate results, operate on all steps. Needing to learn a new strategy for remembering and integrating partial results may outweigh the benefits of computing partial results in the same way. This explanation would also account for the benefit of using an old integrative structure on executing new component steps. According to the second explanation, the benefit of using old component steps in a new integrative structure may have been offset by just having to retrieve the old component
calculations in a new order. These alternatives were explored in experiments described in the following sections.

TWO-CONTEXT TRAINING

Experiment 1 provided a baseline of how integrative and component knowledge transfers to a new procedure when practiced in one training procedure. Because both component and integrative knowledge transferred to a new procedure, extended practice did not render the knowledge completely unavailable, as an extreme application of composition might suggest. Practicing the same knowledge in the context of different procedures may still lead to increased transfer, relative to a single-context training situation like Experiment 1. The hypothesis is that two procedures that share some identical steps may access a single, procedure-independent representation of that information in memory.

General Method

In the two-context experiments, subjects memorized and practiced two mental arithmetic procedures in the first half of each session. These two procedures had one of two features in common. They either had the same integrative structure, with different component steps, or they had the same component steps, with different integrative structures. Subjects alternated between these two training procedures to solve problems.

For the transfer half of the session, they continued to use one of these procedures without any changes. The other procedure they had initially practiced was changed according to one of the three transfer conditions used in Experiment 1: It kept the same integrative structure, but had new component steps; it kept the same component steps, but had a new integrative structure; or it was an entirely new procedure that had both new component steps and a new integrative structure. Of the two training procedures, this will be referred to as the changed procedure. After learning the changed procedure, subjects continued alternating between the two procedures to solve problems in the second half of the session. Thus, these manipulations of the changed procedure constitute a replication of Experiment 1's design. The only difference is that the changed procedure was initially learned and practiced in the context of another procedure that had some similar features.

The materials, instructions, and procedure for Experiments 2-5 were similar to those described for Experiment 1. Subjects learned two procedures and alternated between them to solve 70 problems. In Experiment 1, transferred steps had received 50 practice trials in a single context. To have the same total trials on the common steps present in both procedures, 25 trials
per procedure would have been appropriate. However, pilot subjects did not reach the same level of performance on either procedure, given 25 trials each, as they had on Experiment 1's single procedure after 25 trials. To equate the absolute number of trials on a given procedure, subjects would need 100 total trials before transfer and 100 afterwards. Having 70 learning trials (35 per procedure) and 70 transfer trials represented a compromise between long subjects needed to become skilled at two procedures and how long a reasonable experimental session could last.

Shared Component Experiments

Experiment 1's results indicated that there may be some cost associated with transferring old component steps but changing their order within the new procedure. Experiments 2-4, the three shared component experiments, attempted to disentangle the hypothesized benefit of two-context training from the influence of step-order knowledge. In each of these experiments, the two training procedures had the same component steps used within different integrative structures. The experiments differed according to whether these same steps appeared in the same or different orders during training and after transfer.

Experiments 2 and 3. In both Experiments 2 and 3, the two training procedures shared common component steps but used them in two different orders. Experiment 2 and 3 differed according to whether the shared component steps appeared in an old order or in a completely new order when they were transferred. In Experiment 2's transferred-component condition, transferred-component steps appeared in one of the old orders. In Experiment 3, the transferred-component steps appeared in a new order. When a new integrative structure was needed (for the transferred-component condition and the control condition), it had the same relative ordering of integrative and component steps as the structure it replaced.

Experiment 4. This experiment was designed to assess the benefit of holding step order constant during training on subsequent transfer. The order of the component steps in the two training procedures was the same. The two procedures also used the same relative ordering of component and integrative steps. Thus, in both procedures, the shared-component steps appeared in exactly the same places. When these shared-component steps were transferred to a new procedure, they appeared in a different order.

Shared Integrative Experiment

Experiment 5. Experiment 5 investigated whether practicing an integrative procedure in the context of two procedures affected its availability for transfer to a new procedure. The general method was the same as the pre-
previous two-context learning experiments. The two initially learned procedures in Experiment 5 shared the same integrative structure, but used different component steps. For half the subjects, the first two procedures used the mixed ordering of integrative and component steps. For the other subjects, the two procedures used the blocked ordering of integrative and component steps. The transferred-component and control conditions needed a new integrative structure for the changed procedure. For these conditions, if the integrative structure in the two training procedures used blocked-step ordering, the changed procedure's new integrative structure used mixed ordering. If the training procedures used mixed ordering, then the changed procedure's new integrative structure had blocked-step ordering.

Overview of Analyses

There are two different types of data that should reflect how shared knowledge is represented. The first is the effect of practice on the time to execute shared steps during initial learning. The second is the transfer performance, measured in the same way used in Experiment 1. The practice-effect analyses examined changes in the time to execute steps shared by two procedures. Therefore, they used only the pretransfer execution times for the two training procedures. These reaction-time data were fit to models that assumed that the two shared steps had either (1) a single, procedure-independent representation, or (2) two procedure-specific representations.

The success of the models, combined with a straightforward interpretation of the composition assumption, should in turn predict the transfer patterns. Transfer was measured for each two-context experiment in the same way used for Experiment 1. If the models support a procedure-independent representation of shared steps, then transfer should be greater in the two-context experiments relative to the single-context experiment. If the practice models support procedure-specific representations of common steps, then the amount of transfer should not be different than the amount found in Experiment 1.

Practice Effects During Training

Modeling Practice Learning with a Power Law. In all experiments, subjects' time to execute the training procedures decreased considerably with practice, initially averaging about 10 s per step and finishing at around 2.5 s per step. Newell and Rosenbloom (1981) have demonstrated that practice learning in a variety of tasks can be described by a power law, such as

$$T = A + BP^{-c}$$

(1)

where $T$ is time, $A$ is a measure of asymptotic performance, $B$ is a measure of initial time, $P$ is practice, and $c$ is the rate of change. For the analyses
that follows, a power function is assumed to be an appropriate model of
learning through practice, because of its ability to fit a variety of empirical
learning curves. Once it is established how well a power law fits the learning
data for this particular task, the analysis will focus on how the relative
values of the parameter estimates change as a function of shared knowledge.

Parameter values for Equation 1 were found so that the predicted data
values were close to the actual data values. In order to assess whether the
differences between the predicted and actual data were due to chance, the
differences were squared and summed. There are two reasons why the re-
sulting number could be large. First, it could be large because the model
does not fit the data well. Second, it could be large because there is random
fluctuation in the data. To take the second factor into account, these sums
of squared deviations were divided by an estimate of the variance. This
goodness-of-fit measure, also known as a chi-square statistic, can then be
compared across different sets of data in different experiments. Chi-squares
are evaluated relative to their degrees of freedom (df). In this case, the
degrees of freedom are the number of predicted data points minus the
number of model parameters. For Experiment 1, there were 10 predicted
data points (50 total trials divided into 10 sets of 5 trials each), so $df = 7$. For
Experiments 2-5, there were 7 predicted data points in each experiment (35
trials per procedure divided into 7 sets of 5 trials each), so $df = 4$.

Table IV summarizes the individual fits of integrative and component
step times for the training procedures to Equation 1. As expected, the data
were fit well by a power-law function, with the exception of Experiment 1
component steps and Experiment 3 integrative steps. In each of these two
cases, the deviations were limited to two poorly fit data points. There is
some variability in the parameter estimates. For example, the B parameter
estimates for shared component steps ranges from 3.7 to 6.8 in the three
shared-component experiments. The largest disparity occurs for Experi-
ment 2 and 3, which had identical training situations. For this case, the
variation is probably due to sampling error.

The important feature of the power-law fits is how the parameter esti-
mates change as a function of single-context versus two-context learning.
The two training procedures in Experiments 2-4 shared the same component
steps, so these were the two-context learning situations for component steps.
Along with Experiment 1, Experiment 5 counts as a single-context exper-
iment for component steps because there was a different set of component
steps in each procedure. Conversely, Experiments 1-4 count as single-con-
text experiments for integrative structures. Experiment 5 was the only two-
context integrative experiment.

1 For all practice effect models described here, parameter values were estimated using a
steepest-descent algorithm to minimize chi-squares.
TABLE IV
Parameter Estimates for Model $T = A + B \times P^{-c}$ of Initial Learning Data for Experiments 1–5

<table>
<thead>
<tr>
<th>Parameter</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>$\chi^2$</th>
<th>df</th>
</tr>
</thead>
<tbody>
<tr>
<td>Experiment 1: Single context Component</td>
<td>2.28</td>
<td>8.59</td>
<td>0.98</td>
<td>11.44</td>
<td>7</td>
</tr>
<tr>
<td>Integrative</td>
<td>1.49</td>
<td>6.43</td>
<td>1.19</td>
<td>6.97</td>
<td>7</td>
</tr>
<tr>
<td>Experiment 2: Shared component Component</td>
<td>1.39</td>
<td>3.76</td>
<td>1.10</td>
<td>2.99</td>
<td>4</td>
</tr>
<tr>
<td>Integrative</td>
<td>2.00</td>
<td>6.50</td>
<td>0.85</td>
<td>2.99</td>
<td>4</td>
</tr>
<tr>
<td>Experiment 3: Shared component Component</td>
<td>2.60</td>
<td>6.80</td>
<td>0.95</td>
<td>1.00</td>
<td>4</td>
</tr>
<tr>
<td>Integrative</td>
<td>1.59</td>
<td>5.70</td>
<td>0.82</td>
<td>12.44</td>
<td>4</td>
</tr>
<tr>
<td>Experiment 4: Shared component Component</td>
<td>2.63</td>
<td>4.86</td>
<td>1.69</td>
<td>1.49</td>
<td>4</td>
</tr>
<tr>
<td>Integrative</td>
<td>0.01</td>
<td>4.30</td>
<td>0.59</td>
<td>2.99</td>
<td>4</td>
</tr>
<tr>
<td>Experiment 5: Shared integrative Component</td>
<td>2.23</td>
<td>10.70</td>
<td>0.93</td>
<td>1.00</td>
<td>4</td>
</tr>
<tr>
<td>Integrative</td>
<td>0.01</td>
<td>4.38</td>
<td>0.59</td>
<td>2.99</td>
<td>4</td>
</tr>
</tbody>
</table>

The way in which parameter estimates change according to this manipulation will have implications for alternative models of knowledge representation. If the two similar procedures accessed a single representation of their common steps, the amount of practice for those steps should in effect be doubled. Doubling the amount of practice has its effect on the value of the $B$ parameter:

$$T = A + B_1(2P)^{-c}$$

$$= A + B_2P^{-c}, \text{ where } B_2 = B_1/2^c$$

When $c$ is about 1.0, the $B_1$ parameter, used for two-context experiments, should be about half the size of the $B_1$ parameter used for one-context experiments. When $c$ is somewhat less than 1.0, then $B_2$ will still be substantially larger than $B_1$.

As Table IV indicates, the $B$ estimates for component step learning in two contexts (Experiments 2–4) are considerably smaller than the estimates for the single-context component experiments (Experiments 1 and 5). On average, this relationship is quite close to what would be expected from Equation 2. Similarly, the value of $B$ for integrative steps is smaller in the

\[ \chi^2 \]

$\chi^2$ The chi-square was calculated by dividing the sum of squared deviations by an average measure of the variance associated with the practice effect across the two-context experiments.
two-context integrative experiment (Experiment 5) than in the single-context integrative experiments (Experiments 1–4). In both the integrative and component cases, the relationship between the B parameters in Table IV is at least consistent with the notion that common steps were receiving the benefit of 70, not 35, practice trials.

**Evaluating Alternative Knowledge Representation Models.** To get a more precise evaluation of this shared-representation assumption, three different models were fit to the learning data. Model 1, the "single-representation" model, assumes something different happens when steps are practiced in two contexts than in one context. Therefore, it uses two different B parameters to fit the data. Model 2, the "strict single-representation" model, assumes that a single representation of shared steps received twice as much practice in the two-context experiments than in the single-context experiments. In this case, the $B_2$ parameter should not be just different from $B_1$, but should have the relation given in Equation 2: $B_2 = B_1/2$. The "separate representations" model, Model 3, assumes that steps practiced in two procedures have a different representation associated with each procedure. There should be no difference between steps practiced in two procedures and steps practiced in one procedure. They should be affected only by the amount of practice given to the procedure in which they occurred. This model fit the data from both single-context and two-context experiments with a single B parameter.

There is not enough knowledge about performance on this kind of task to interpret absolute values of parameter estimates. For Model 1, however, the B parameters for the single- and two-context learning experiments should have the relationship given in Equation 2. Models 2 and 3 represent additional constraints on the B parameter. If one of these models fit the data well, relative to Model 1’s fit, this would support the model’s constraint and its associated knowledge representation assumption. If the fit were significantly worse than Model 1’s fit, then this would be evidence against the constraint.

Model 1 was represented as two power-law functions that differed in their B parameters:

\[ T = A + B_1 P^{-c} \]  
(3)

\[ T = A + B_2 P^{-c} \]  
(4)

The asymptotic level of performance (the A parameter) was assumed to vary across experiments. The first set of model fits concerned component-step performance. Component-step data from Experiments 2, 3, and 4 were fit to Equation 3. A common exponent was used to fit data from all three ex-
periments. This is a stringent test for the model because each experiment had different subjects and potentially different variances. As Table IV illustrates, there was a good deal of variation in the exponent estimates (.95 to 1.69), so this approach would only serve to hurt the degree of fit. Component-step data from Experiment 5, the single-context case for component steps, were fit to Equation 4. Thus, seven parameters were estimated, given 28 data points: 4 values of A, a value for B₁, a value for B₂, and a value for c. Table V gives the parameter estimates and chi-squares for this model.

### Table V
Parameter Estimates for Three Models of Two-Context Practice Learning

<table>
<thead>
<tr>
<th>Parameter Estimates</th>
<th>A₁</th>
<th>A₂</th>
<th>A₃</th>
<th>A₄</th>
<th>B₁</th>
<th>B₂</th>
<th>C</th>
<th>χ²</th>
<th>df</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Component Steps</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Model 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T = A₁ + B₁ X P⁻ᵣ, i = 2, 4</td>
<td>2.53</td>
<td>3.10</td>
<td>1.54</td>
<td>2.64</td>
<td>10.37</td>
<td>6.14</td>
<td>1.02</td>
<td>40.30</td>
<td>21</td>
</tr>
<tr>
<td>Model 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T = A₁ + B₁ X P⁻ᵣ, i = 5</td>
<td>2.11</td>
<td>2.58</td>
<td>1.11</td>
<td>1.62</td>
<td>11.46</td>
<td>-</td>
<td>0.85</td>
<td>57.58</td>
<td>22</td>
</tr>
<tr>
<td>Model 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T = A₁ + B₁ X P⁻ᵣ, i = 2, 5</td>
<td>2.21</td>
<td>2.69</td>
<td>1.12</td>
<td>3.85</td>
<td>7.15</td>
<td>-</td>
<td>1.05</td>
<td>414.02</td>
<td>22</td>
</tr>
<tr>
<td><strong>Integrative Steps</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Model 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T = A₁ + B₁ X P⁻ᵣ, i = 2, 4</td>
<td>0.95</td>
<td>1.69</td>
<td>0.50</td>
<td>5.85</td>
<td>3.90</td>
<td>0.74</td>
<td>119.20</td>
<td>21</td>
<td></td>
</tr>
<tr>
<td>Model 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T = A₁ + B₁ X P⁻ᵣ, i = 2, 4</td>
<td>0</td>
<td>1.34</td>
<td>1.67</td>
<td>0.70</td>
<td>5.89</td>
<td>-</td>
<td>0.75</td>
<td>124.38</td>
<td>22</td>
</tr>
<tr>
<td>Model 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T = A₁ + B₁ X P⁻ᵣ, i = 2, 5</td>
<td>3.26</td>
<td>1.69</td>
<td>2.02</td>
<td>0</td>
<td>5.26</td>
<td>-</td>
<td>0.76</td>
<td>189.05</td>
<td>22</td>
</tr>
</tbody>
</table>

*The value of i indicates which two-context experiments were fit to the model.*

Under this model, B₁ was estimated at 10.37. From Equation 2 B₁ should equal 5.11 (10.37 / 2.02). Its estimated value, 6.14, is only somewhat larger. Because the range in the B parameter was quite large when estimated for each experiment separately (see Table IV), a 20% deviation from the model's predicted value indicates a relatively good fit. Generally, the model fit the data well by assuming something different happens depending on whether steps are practiced in one context or two.

Model 2, the strict single-representation model, assumes that the same component steps received twice as much practice when they appeared in two procedures (Experiments 2-4) than when they appeared in a single procedure (Experiment 5). It uses a single B parameter to capture this:
The component step data for Experiments 2–4 were fit to Equation 5; the component data from Experiment 5 were fit to Equation 6. The main finding in Table V is that the chi-square increased only slightly under this model. (The degrees of freedom increase for this model because there is one less parameter.) This supports the shared representation assumption, especially when contrasted with Model 3’s performance. For Model 3, the separate-representations model, data from all four two-context experiments were fit to Equation 6. The chi-square, 414.02 ($df = 22$), was substantially larger than the Model 2 chi-square, $F(22,22) = 7.19, p < .001$. In general, the separate-representations model did a worse job in fitting the data than either of the shared representation models.

Figure 2. Experiments 4 and 5 component step execution time during training and predicted times from Model 2 (shared representation) and Model 3 (separate representations).
All the chi-squares for these three models are significantly larger than chance. This is not surprising because they attempted to fit data from different experiments that involved different subjects using a single practice parameter and only two initial-time parameters. The important point is not the absolute fit of the alternative models, but their relative ability to capture the learning data. Figure 2 illustrates the fit of predicted to actual data under Models 2 and 3 for Experiment 4 (shared component steps in the two training procedures) and Experiment 5 (different component steps in the two training procedures). Although Model 2, the shared-representation model, deviates from the actual data, particularly on later trials, it clearly does a better job of capturing the nature of practice effects across the two learning situations.

A similar set of model fits was done with integrative step data. Experiment 5's training procedures shared the same integrative structure, so its in-
tegrative step data were fit to Equation 3 for Model 1. For Experiments 2-4, the training procedures had different integrative structures, so their data were fit to Equation 4. Under Model 1, $B$, was estimated at 5.85. From Equation 2, the value of $B_t$ should be 3.50 ($5.85 \times 2^{-0.74}$); it was estimated at 3.90 (see Table V).

This kind of agreement supports the notion that a single representation of shared integrative information was accessed by both training procedures. However, this conclusion must be qualified by the poor fit by all three models, as evidenced by the large chi-square values in Table V. This can also be seen in Figure 3, which illustrates the actual and predicted data for integrative steps under Models 2 and 3 for Experiments 4 and 5. When different integrative structures must be learned (Experiment 4), both models overestimate practice effects early during learning and underestimate them for later trials. As noted earlier, a power law did not fit Experiment 3's integrative steps very well. This may account in part for the poor performance of all three models, as well as some of the unreasonable asymptote estimates of 0. Conservatively interpreted, these results offer no consistent support for the assumption that a single representation of common integrative knowledge was shared.

**Transfer Effects After Training**

The practice effect data support a single representation of component steps shared by two different procedures. Given these results, transfer of component steps in the shared component step experiments (Experiments 2-4) should be greater than the transfer found in Experiment 1. The practice effect analyses provide less compelling evidence for a single representation of a shared integrative structure. Therefore, no significant transfer benefit is expected for shared integrative steps in Experiment 5 relative to the amount found in Experiment 1.

To assess the impact of two-context training on transfer, the amount of savings found for transferred knowledge in each two-context experiment will be contrasted with the amount of savings found for transferred knowledge in Experiment 1. The power of the cross-experimental tests will be low because the subjects were different from experiment to experiment and there is a great deal of variance. Although differences in the right direction may occur to support the hypothesis, they may not be statistically reliable. To be conservative, only statistically reliable differences will be taken as evidence in favor of the hypothesis that two-context training aids transfer.

**Experiments 2 and 3.** The shared component steps in these experiments appeared in different orders in the training procedures. In Experiment 2, the transferred component steps retained one of these old orders. In Experi-
ment 3, they appeared in a completely new order. If step-order information has no impact on the transfer of component steps and if a procedure-independent knowledge structure is more available for transfer, then both experiments should show greater component-step transfer than Experiment 1. If step-sequencing information is critical to component-step transfer, then Experiments 2 and 3 should have different patterns of results.

The mean reaction times for the changed procedure and the differences relative to the control condition are given in Table VI. For the two-context experiments, the mean component step times are based on the first 25 practice trials with the changed procedure. These 25 trials really reflect 50 total practice opportunities because these same steps were used in the unchanged procedure as well. These 50 practice trials, experienced across two contexts, must be compared to the 50 trials experienced in one context in Experiment 1. From Experiment 1, the baseline difference between the transferred component and control conditions was 0.62 s. In Experiment 2, the advantage of transferring old component steps was about three times as great.

There are two candidate explanations for this increased transfer in Experiment 2: (1) The increased availability afforded by the procedure-independent representation of component steps, or (2) the use of transferred component steps in a familiar order, which had not been the case in Experiments 2-5

<table>
<thead>
<tr>
<th>Condition</th>
<th>Component</th>
<th>Integrative</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>M</td>
<td>CTD</td>
</tr>
<tr>
<td>Experiment 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Control</td>
<td>5.06</td>
<td>2.68</td>
</tr>
<tr>
<td>Transferred integrative</td>
<td>4.58</td>
<td>2.39</td>
</tr>
<tr>
<td>Transferred component</td>
<td>3.17</td>
<td>2.36</td>
</tr>
<tr>
<td>Experiment 3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Control</td>
<td>4.42</td>
<td>2.52</td>
</tr>
<tr>
<td>Transferred integrative</td>
<td>4.46</td>
<td>2.06</td>
</tr>
<tr>
<td>Transferred component</td>
<td>3.85</td>
<td>2.46</td>
</tr>
<tr>
<td>Experiment 4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Control</td>
<td>5.15</td>
<td>2.85</td>
</tr>
<tr>
<td>Transferred integrative</td>
<td>5.09</td>
<td>3.01</td>
</tr>
<tr>
<td>Transferred component</td>
<td>4.18</td>
<td>3.12</td>
</tr>
<tr>
<td>Experiment 5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Control</td>
<td>5.09</td>
<td>3.13</td>
</tr>
<tr>
<td>Transferred integrative</td>
<td>4.94</td>
<td>1.71</td>
</tr>
<tr>
<td>Transferred component</td>
<td>4.61</td>
<td>2.92</td>
</tr>
</tbody>
</table>
The results from Experiment 3 support the second interpretation. The 0.57 s savings for transferring old component steps—but using them in a new order—was not statistically reliable (see Table VI).

Taken together with Experiment 1, Experiments 2 and 3 indicate that the degree of component step transfer is influenced by retaining step order. Changing the order of transferred steps detracts from transfer. Perhaps varying the order of shared steps during learning reduced the level of initial skill with the steps. If the order of shared steps is constant across the training procedures, then subjects may reach a higher level of initial skill. This in turn may lead to better subsequent performance, when step order is changed during transfer. Experiment 4 tested this hypothesis.

Experiment 4. The shared component steps in Experiment 4 appeared in the same order and on the same steps in both training procedures. They were used in a different order when transferred to the changed procedure’s new integrative structure.

Table VI presents the data for the changed procedure for the first 25 transfer trials. There was a reliable benefit (0.97 s) for using old component steps, $F(1,17) = 14.4$, $p < .002$. Practicing the same component steps in precisely the same order helped initial learning as well. Prior to transfer, Experiment 4 subjects averaged well over a second faster on component step times than subjects in Experiments 2 and 3. However, the knowledge associated with these steps was no more available for new procedures than if it had been practiced on just a single procedure. Although the 0.97 s advantage was about 56% larger than the Experiment 1 difference for the same number of practice trials, this difference in the degree of transfer was not statistically reliable. Taken together, these three shared-component step experiments indicate that the order for using component steps impacts both learning and transfer. Holding step order constant during training ensures better learning, but not better transfer if that order is altered.

Experiment 5. In Experiment 5, the two training procedures had a common integrative structure but used different component steps. Table VI shows there was a significant benefit, 1.42 s, for transferring shared integrative steps in the changed procedure. This savings was about 58% greater than the savings associated with transferred integrative steps in Experiment 1 (0.90 s). However, this amount of transfer was not significantly larger than the amount found for transferred integrative steps in Experiment 1. On this basis, the results indicate that practicing on integrative structure in the context of two procedures did not reliably increase transfer relative to single-procedure practice.

*The difference between the control-transferred-component savings between experiments was evaluated using a $t$-test for small samples recommended by Winer (1971).*
Summary of Practice and Transfer Results

All experiments demonstrated that knowledge associated with component and integrative steps transferred to new procedures. However, component step transfer was consistently influenced by aspects of the integrative structure and steps. This was demonstrated most convincingly by Experiment 1's result that, over time, subjects were faster at executing new component steps in an old integrative structure than they were at executing old, well-practiced component steps in a new integrative structure.

The implications of the practice effect analyses and the transfer results for the two-procedure experiments are inconsistent. The practice effect models supported a procedure-independent representation of common knowledge, and therefore would predict increased transfer for component steps. Although the interexperimental comparisons were in the right direction, the advantage was not strong enough to be convincing. The two-context experiments did illustrate that changes to step order affected the degree of transfer. Holding the step order of component steps constant from initial learning to transfer had a clear advantage relative to any situation in which these steps were reordered. Likewise, there was a substantial benefit to initial learning when identical component steps occurred in the same order across two otherwise different procedures. Yet this benefit did not carry over to transfer if these calculations were reordered for a new procedure. A model of how procedural knowledge was represented for this task must resolve the apparent contradiction of procedure-independent knowledge structures that did not always show greater transfer than procedure-specific knowledge structures.

GENERAL DISCUSSION

This study investigated two issues of procedural knowledge. First, how are similar well-learned procedures represented? Second, what implications does their representation have for a composition-like mechanism? Two assumptions were made. The first was that something like composition underlies the quantitative and qualitative features of skilled performance that occurs after considerable practice. The second was that the component/integrative knowledge distinction was both a real and important feature of cognitive skills. Both assumptions were consistent with the results obtained. As it happens, the component/integrative distinction plays a major role in reconciling the evidence for a procedure-independent representation of similar parts with the lack of a corresponding transfer benefit.

Skilled performance on this task can be characterized as getting the correct answer as fast as possible. To do this, a person must recall which calculation to do, have available any necessary intermediate results, and
know how to execute the calculation. The first two types of information have been associated with the skill's integrative framework. This framework organized and integrated intermediate results. The intermediate results were provided by the third type of information, the calculations. The integrative structure affected component step times in two ways. First, new component steps were executed faster in an old integrative structure than in a new integrative structure. Second, changing the integrative structure disrupted performance on old, well-practiced component steps. These results rule out some models of how knowledge was represented for this skill. For example, one plausible model might propose that (a) all intermediate results were stored in serial order while the procedure was being executed, and (b) the knowledge about which results to retrieve and when to combine them was localized to the integrative steps. However, it would be difficult for this model to account for the integrative structure's effects on component step execution. In some cases, all the component steps were completed before a single integrative calculation was performed. If integrative steps were the sole repository of integrative knowledge, there would not have been any influence of integrative features on component step performance.

The results reinforce the initial characterization of this task's integrative structure and support an alternative model of its representation. Integrative knowledge in this task was not just the integrative calculation or localized to the integrative steps. Skilled performance on this task required both integrative level knowledge and component level knowledge on each step. The model proposed here claims that two types of integrative information are used at each step. One is the association of a particular step with a particular calculation. More generally, this corresponds to, “What do I do now?” knowledge. The results of Experiments 2–4 support the impact of this information for both initial learning and transfer. Can this, “What do I do now?” knowledge alone account for the influence of the integrative structure on component step times? Given Experiment 1's transfer patterns, the answer is no. New component steps in an old integrative structure were executed faster than new component steps in a new integrative structure. In each situation, the what-to-do-now knowledge as well as the details of the calculation were different on component steps. What transferred in the first situation was knowledge about when and how those results would subsequently be needed. Thus, both knowledge about what to do and knowledge about how the result would be subsequently used comprise the integrative information associated with each step of the procedure.

This integrative structure forms a context for each step of the procedure. The integrative calculations can be thought of as a kind of action template specifying a known operation or action (fixed) and two arguments (changing from problem to problem). The two arguments come from intermediate results calculated on preceding steps. The integrative kno-
edge for those preceding steps would indicate which templates, used later in the procedure, need the current result. Thus, the information being maintained in memory is not intermediate results, but rather, action templates in various stages of completion. When an integrative calculation is reached, the template's action can be applied to the waiting arguments.

This model can reconcile results supporting a procedure-independent representation of component calculations with a small transfer benefit. If time measured on component steps reflects the integrative context as well as the knowledge about the specific calculation, then changing that context must impact total step time. This will be true regardless of whether the knowledge for achieving the intermediate result has a procedure-independent representation.

Although the practice effect models supported a single-representation of common component steps, the evidence was less compelling for a single-representation of a common integrative structure. All practice effect models were quite poor at predicting integrative step times. There are a few possible reasons why this occurred. First, it is clear that an important part of the integrative knowledge was operating on component steps. However, the practice effect models used only the integrative step times. The second reason has broader implications for understanding how a composition mechanism works on complex cognitive procedures. The integrative structure specified what the integrative demands were, but not how to accommodate those demands. There is considerable room for strategy variation within a person's approach to the task. The representative of integrative knowledge may have undergone strategic transformations as subjects become more familiar with the task. This would be consistent with the finding that none of the power-law models did a particularly good job at fitting the integrative step data. However, a composition mechanism that bundles up sequentially executed steps cannot yield strategic kinds of transformation (Lewis, 1981). Because each transformation constitutes a different internal representation of integrative knowledge, these transformations present a moving target for an automatic composition mechanism attempting to collapse sequentially accessed knowledge. Although composition predicts an exponential speed-up given otherwise static knowledge structures (Lewis, 1978), perhaps it rarely has an opportunity to operate on such structures. This would be particularly true in early stages of skill acquisition when other strategic transformations play an important role.

The view that intermediate states or results of a procedure are represented in terms of their subsequent roles is consistent with other transfer-of-training results. In quite a different domain, for example, Thomas (1974) examined problem-solving transfer using the hobbits and orcs problem. This problem has a long solution path and naive subjects characteristically make errors in the middle. Some subjects had prior training on the second
half of the solution path, using a mid-solution state as the starting state. When given the whole problem, these subjects did not do any better on the second half of the solution path than they did on the novel first half. Nor did they have fewer errors on the midsolution state (their starting state during training) than subjects who had no prior exposure to the problem. According to the model outlined above, the lack of positive transfer on the midsolution state and the second half of the solution path resulted from their use in two different integrative contexts. As the starting state during training, characteristics of the midsolution state were used to generate a series of moves. In its usual position along the solution path, it more likely occurred in a chain of moves planned from some other state. Although the set of solution-path moves from this state are the same in both cases, the contribution of those moves to a solution strategy is probably different.

In summary, this study has shown that similar cognitive procedures will share a single representation of their common parts, at least when the procedures are learned at the same time. However, the transfer benefit one might expect from this procedure-independent representation does not always occur. The strategic or integrative types of knowledge associated with cognitive procedures may influence (a) how "transportable" procedure parts are, and (b) how successful a composition-like mechanism will be at transforming knowledge structures as a function of practice. There may not be much net transfer for familiar "local" routines to new tasks if those tasks impose new integrative contexts for using them.

The evidence that different procedures may access a single representation of their identical subparts is comforting from a cognitive economy viewpoint. However, it is not clear whether this task encouraged such economy or whether similar parts were abstracted from redundant, procedure-specific representations. The second possibility is critical because people are more likely to acquire similar skills sequentially, rather than in parallel. How a mechanism might operate for recognizing similarities between a well-entrenched procedure and a new one, and whether an economical "sharing" of similar parts would still occur, are important issues for theories of procedural knowledge.

REFERENCES


