Cognitive Psychology **42,** 267–316 (2001)
doi:10.1006/cogp.2000.0747, available online at http://www.idealibrary.com on IDEAL®

Does Learning a Complex Task Have to Be Complex?: A Study in Learning Decomposition

Frank J. Lee and John R. Anderson

Carnegie Mellon University

Many theories of skill acquisition have had considerable success in addressing the fine details of learning in relatively simple tasks, but can they scale up to complex tasks that are more typical of human learning in the real world? Some theories argue for scalability by making the implicit assumption that complex tasks consist of many smaller parts, which are learned according to basic learning principles. Surprisingly, there has been rather sparse empirical testing of this crucial assumption. In this article, we examine this assumption directly by decomposing the learning in the Kanfer–Ackerman Air-Traffic Controller Task (Ackerman, 1988) from the learning at the global level all the way down to the learning at the keystroke level. First, we reanalyze the data from Ackerman (1988) and show that the learning in this complex task does indeed reflect the learning of smaller parts at the keystroke level. Second, in a follow-up eye-tracking experiment, we show that a large portion of the learning at the keystroke level reflects the learning even at a lower, i.e., attentional level. © 2001 Academic Press

Over the past 2 decades there have appeared a number of theories of skill acquisition (Anderson, 1982; Delaney, Reder, Staszewski, & Ritter, 1998; Logan, 1988; Mackay, 1982; Rickard, 1997) that have had considerable success in addressing the fine details of learning relatively simple tasks. While encouraging, these demonstrations leave the worry that there may be problems with scaling up to the complex tasks that are more typical of human learning in the real world. A challenge to the learning theories is whether they can provide a complete characterization of the acquisition of truly complex skills. In the limit, this is an impossible ambition. By its nature, any very complex skill is going to be so knowledge intensive that a complete characterization (i.e., tracing out all of the components of that knowledge) will overwhelm the capacities of any researcher. However, a reasonable goal

The research reported in this article was supported by the Office of Naval Research, Cognitive Science Program, under Contract No. N00014-95-1-0223 to John R. Anderson. Address correspondence and reprint requests to Frank J. Lee, Department of Psychology, 342-C Baker Hall, Carnegie Mellon University, Pittsburgh, PA 15213-3890. E-mail: fjl+@cmu.edu or ja+@cmu.edu.

0010-0285/01 $35.00

Copyright © 2001 by Academic Press
All rights of reproduction in any form reserved.

for learning theories is that they try to address more complex tasks and strive for a more complete characterization of these tasks.

To foster this goal the Office of Naval Research (ONR) designated several tasks, including the Kanfer–Ackerman Air-Traffic Controller (KA-ATC) Task (Ackerman & Kanfer, 1994), as challenge tasks for the Hybrid Architectures Project. In the case of the KA-ATC task, Ackerman (Ackerman & Kanfer, 1994) has made available through ONR the detailed data from 11 experiments. In this article we describe an analysis of the data from 1 of these experiments and a follow-up eye-tracking experiment designed to elucidate the nature of the learning in this task. We argue that the learning at the global level reflects the learning of the lower level components. In particular, we argue that much of the learning in this task is driven by changes in attentional strategies in which fixations of task-irrelevant information are reduced as a function of practice. But before we begin, we briefly discuss the power law of practice (Newell & Rosenbloom, 1981).

THE PRACTICE FUNCTION

In their article on the power law of practice, Newell and Rosenbloom (1981) surveyed a large range of skills which appeared to speed up according to a power function and ranged in scale from less than 1 s up to 30 min at the beginning of the task. A power function relates the time (T) to perform a task to the amount of practice by a function as follows:

$$T = A + BN^{-c},$$

where A is the asymptote, B (which we call the scale factor) is the amount of time that improves with N number of practices, and c is the exponent that specifies the rate of the speed up with practice. A power function decreases from an initial time of $A + B$ to A in the limit. To illustrate the power law with an example, in Fig. 1 we plot the data from a procedure learning task from Anderson, Fincham, and Douglass (1999) and a power function fit to that data. As can be seen, a characteristic of a power function is that it is strongly negatively accelerated. That is, the rate of the speed-up decreases dramatically with each unit of practice. A challenge to theories of skill acquisition is to explain why there is this dramatic deceleration.

We should note at the outset that there is currently a controversy regarding whether these learning functions are really power functions or some other functions, such as exponential functions (cf. Anderson & Tweney, 1997; Wixted & Ebbesen, 1991, 1997; Heathcote, Brown, & Mewort, 2000; Myung, Kim, & Pitt, 2000), or mixtures of different power functions (cf. Rickard, 1997; Delaney, Reder, Staszewski, & Ritter, 1998). Our data does not have anything to say about the controversy between the power–exponent

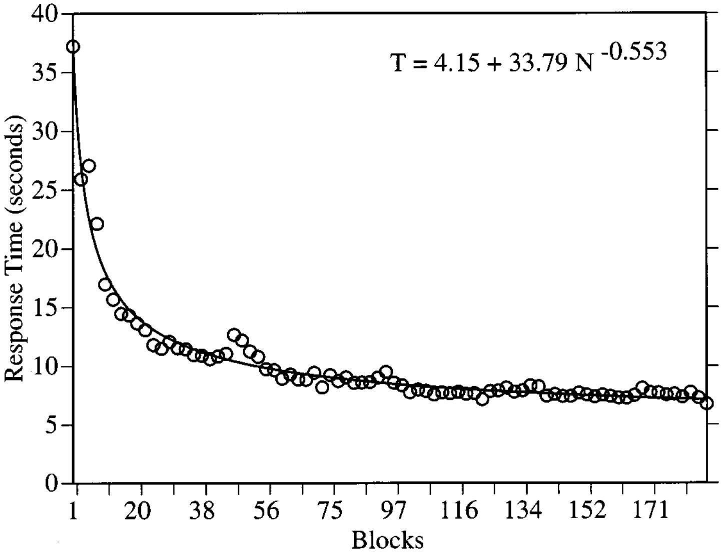


FIG. 1. An example data from Anderson, Fincham, & Douglass (1999) and a power-function fit to that data.

distinction because we are simply using the power function as an approximate descriptive characterization of the learning in the KA-ATC task. Indeed, the arguments presented in this article would not have changed if we had used an exponential or some other learning function. However, we do have something to say regarding the issue of the learning function being a mixture of different power functions. In particular, we suggest that one of the reasons for the observation of the ''approximate'' power-law learning at the global level is because they reflect a mixture of attentional strategies that are being learned at the lower level.

Broadly speaking, theories of skill acquisition can be classified into three categories. The first is the view that skill acquisition results from transforming a procedure, such as when a macro procedure is composed from multistep procedures. Theories that fall under this category include those of Neves and Anderson (1981) and Newell and Rosenbloom (1981). The second is the view that skill acquisition results from strengthening a procedure through strengthening of the individual methods or the connections between the methods that underlie that procedure. Theories that fall under this category include those of Anderson (1982) and Mackay (1982). The third is the view that skill acquisition results from selecting a more speeded method or methods. Theories that fall under this category include those of Crossman (1959); Delaney, Reder, Staszewski, and Ritter (1998); Logan (1988); and Rickard (1997). We discuss each of the categories in more detail below.

Skill Acquisition as Transforming a Procedure

The theories that fall under this category derive at least some—if not all—of their speed-up from the transformation of procedures as a function of practice. A good example of this type of a theory is Newell and Rosenbloom's (1981) chunking theory of learning, in which they proposed that the multistep procedures in performing a task collapsed into a macro procedure as a function of practice. While such step collapsing naturally produces an exponential speed-up, they argued that complex tasks have a combinatorial structure such that while there were many possible macro steps, the learning opportunities for these macro steps decreased with the size of the steps. The net effect of this exponential learning applied to a growing number of macro steps was a power-law learning function. Neves and Anderson (1981) proposed a similar step-collapsing mechanism as part of a larger two-part theory called knowledge compilation. They argued that the speed-up in learning is a reflection of a two-step learning process. In the first step the declarative knowledge of performing a task is proceduralized. In the second step, these procedures are refined through a process called composition, in which multistep procedures are collapsed into a macro procedure.

Skill Acquisition as Strengthening a Procedure

While both Newell and Rosenbloom's (1981) and Neves and Anderson's (1981) theories are well suited to account for learning in complex task domains, Anderson (1982) argued that the power-law learning can be observed for much simpler tasks that did not have a combinatorial structure. Since it is unlikely that a large-scale step collapsing can take place in these simple tasks, Anderson proposed a strength-based learning mechanism that applied to the individual steps of a complex task. He argued that the power-law learning of the complex task was derivative of the power-law learning of the simple steps of a complex task. A similar strengthening mechanism is also at the center of MacKay's (1982) theory. MacKay proposed that knowledge was hierarchically embedded in abstract data structures called nodes. For instance, a node representing a sentence may consist of abstract nodes for the noun phrase and the verb phrase that branched all the way down to the nodes representing the articulators used to sound the syllables of a word in that sentence. He posited a priming mechanism that speeded the responsiveness of other nodes at the same level. He argued that while the priming is itself exponential, the relative abundance of unpracticed nodes, i.e., high-level concept nodes, initially allowed for more learning opportunities in the beginning, but the reduction of these learning opportunities with practice produced a power function speed-up.

Skill Acquisition as Selecting a Procedure

Crossman (1959) formulated one of the earliest theories of power function speed-up. He proposed a theory in which the speed-up in the power law of practice resulted from favoring a faster method in the course of the learning. He posited a stochastic selection mechanism in which the probability of selecting a speedier method from a set of methods increased with practice. He showed that one consequence of such a learning mechanism was a negatively accelerated learning function. However, there were two issues that caused some reservations with Crossman's theory. First, it was unclear what he meant by a method, and second, it was unclear precisely how the method-selection mechanism worked. Logan (1988) clarified these issues within his instance theory of learning by positing two classes of methods, memory retrievals and algorithms, and detailing a selection mechanism in which memory retrievals of the past solutions raced in parallel with an algorithmic solution process. He argued that the power function speed-up reflected the fastest of an increasing number of memory traces. Rickard (1997), however, while accepting Logan's basic distinction between the two classes of methods (memory retrieval versus algorithmic process), was nonetheless critical of Logan's assumption that these two types of methods raced in parallel. Instead, he argued that while algorithmic solutions and retrievals of those solution traces both occurred, they could only occur serially, and the power-law learning at the global level reflected the separate power-law learning of the algorithmic and the retrieval components. Delaney, Reder, Staszewski, and Ritter (1998) forwarded a similar argument. They argued that the systematic deviations from power-function fits that one sometimes saw in the reported literature resulted from a strategy change. They posited that one could achieve a much better fit to the data by fitting separate learning functions for each strategy.

THE REDUCIBILITY HYPOTHESIS

Although the individual theories of skill acquisition have not all explicitly addressed the issue of how to conceive of the learning of complex tasks, the implicit hypothesis that many of them share, which we call the reducibility hypothesis, is that complex tasks simply consist of a lot of little parts that are learned according to basic learning principles. Surprisingly, there has been rather sparse empirical testing of this hypothesis. In much of the literature on learning, a single, global measure of performance is typically measured, analyzed, and reported, regardless of the complexity of the target task. An exception to this is Anderson, Conrad, and Corbett's (1989) analysis of students learning to program in Lisp, in which they decomposed the learning of Lisp programming skills into about 500 production rules. They found that the time to execute these productions speeded up as a function of practice.

While Anderson et al. (1989) provided evidence for the reducibility hypothesis, two aspects of their analysis are causes for reservations. First, there is the worry that Anderson et al.'s reduction of Lisp programming skills to production rules was dependent on the details of their production system theory, which has been subsequently modified (cf. Anderson & Lebiere, 1998). Second, there is also the worry that Anderson et al. did not go far enough in their decomposition. They decomposed learning into units that took from 5 to 50 s to perform. Since learning in many simple tasks is analyzed at the perceptual-motor level, if the reducibility hypothesis truly holds, then one should be able to reduce a complex task down to the perceptual-motor level.

With these concerns in mind, we propose to test the reducibility hypothesis in the KA-ATC task. Recall that the reducibility hypothesis states that complex tasks simply consist of a lot of little parts that are learned according to basic learning principles. To frame it another way, if the reducibility hypothesis is true, then the learning in a complex task can be decomposed into the learning of the smaller components. Therefore, our test of the reducibility hypothesis will consist of (1) hierarchically decomposing the KA-ATC task from the global task level to the elementary keystroke level and (2) examining the data to see if the learning at the lower level is consistent with the learning at the higher level. By consistent learning, we mean that the improvement at the higher level results from the improvement in learning the lower level components. Therefore, learning would not be consistent if there is improvement at the higher level, but some or all of the learning of the lower level components were not improved.

One would not expect to find consistent learning if the main factor that was promoting the learning was conceptual reorganizations reflecting progressive insights into the problem. For instance, consider a person learning how to use Microsoft's Excel program and discovering that one does not have to copy and paste formulas but can simply fill down. At such points, one ought to see new low-level actions entering the person's action sequence and find them taking longer, reflecting their novelty. At the global level, the learning might appear continuous, but at the level of individual actions, there would be discontinuities in the learning. A hypothetical example is provided in Fig. 2. In Fig. 2, an example is provided of learning that is consistent (Alternative 1) and learning that is not consistent (Alternative 2). In Alternative 1, the learning is consistent between the two levels because the learning that we observe at the higher level is reflected in all the lower level components. However, in Alternative 2, the learning is not consistent between the two levels because the learning at the higher level is not reflected in all the lower level components. In particular, one of the low-level components takes longer at the transition point, but the total performance time still speeds up because there are fewer steps after a conceptual reorganization. There is no a priori reason why learning should be consistent between the different levels. It is our hypothesis that learning in a complex task is consistent, i.e., learning at the higher level is reflected in learning at the lower level components.

In this article, we decompose and analyze learning in the KA-ATC task

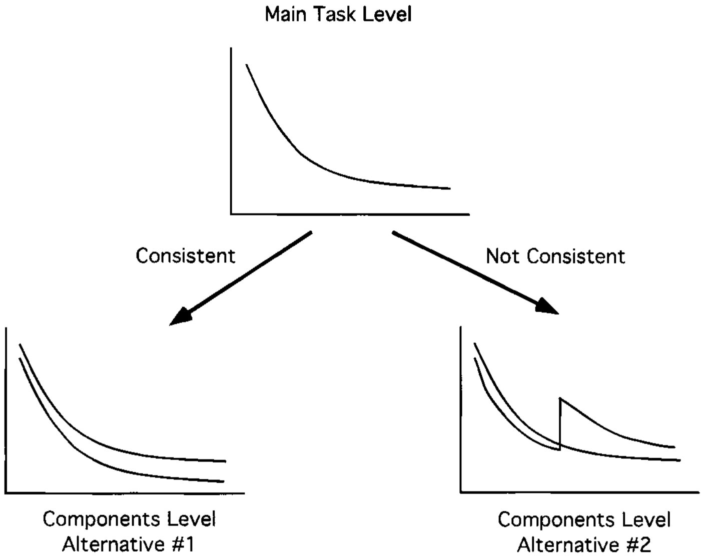


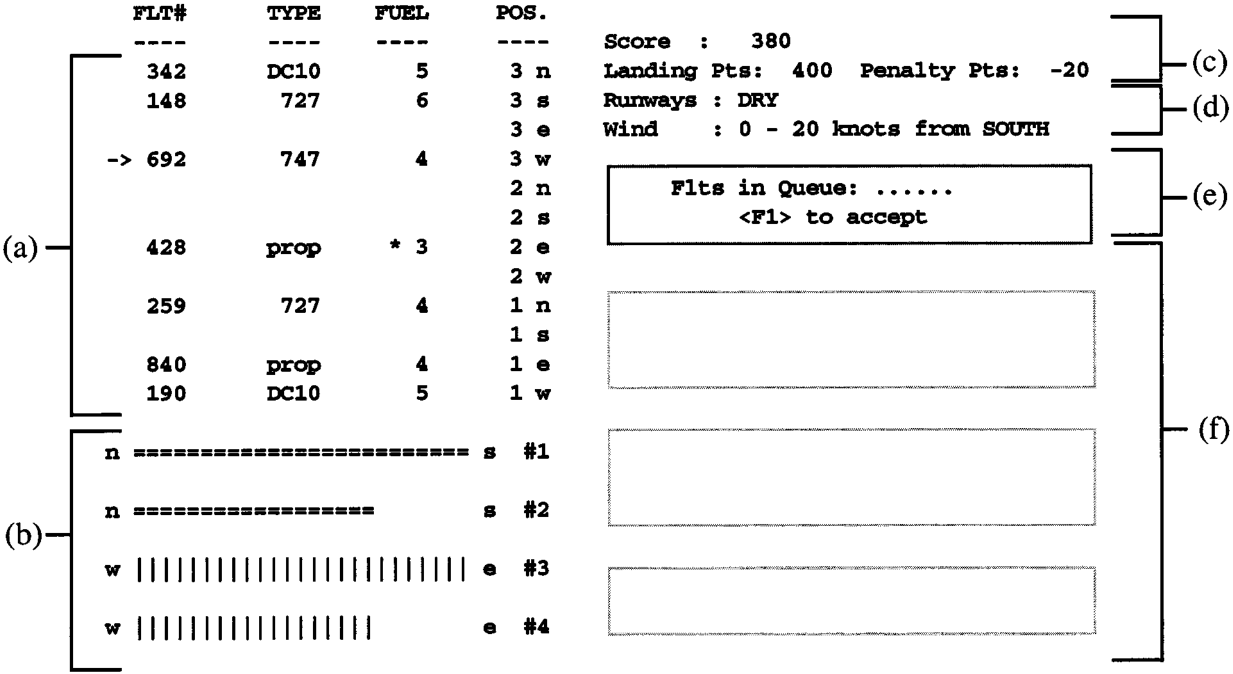
FIG. 2. A hypothetical example of learning that is consistent (Alternative 1) and learning that is not consistent (Alternative 2).

at multiple levels, from the level of overall performance all the way down to the level of keystroke response times. Our method of task decomposition does not depend on any particular theory of skill acquisition. Rather, it solely relies on a pretheoretical task analysis of the KA-ATC task. From our decompositional analysis, we show that the learning at the lower level components is consistent with the learning at the higher levels, thereby providing evidence for the reducibility hypothesis in this task. We show that learning at the lower level involves speed-up of motor actions and changes in where people look.

THE KANFER–ACKERMAN AIR-TRAFFIC CONTROLLER TASK[1]

A typical display of the KA-ATC task is presented in Fig. 3. The KA-ATC task is composed of the following elements displayed on the screen: (a) 12 hold-pattern positions divided into three levels; (b) four runways, numbered 1 through 4; (c) feedback information on participant's current score, landing points, and penalty points; (d) current conditions of the runways, wind direction, and wind speed; (e) a queue of planes waiting to enter the hold pattern; and (f) three message windows, one for notifying of weather

[1] Our description of the KA-ATC task is an amalgamation of the task description provided in Ackerman (1988) and Ackerman and Kanfer (1994). Please refer to those two sources for a more complete account of the task.



```
  FLT#      TYPE     FUEL     POS.
  ----      ----     ----     ----     Score  :   380
   342      DC10        5      3 n     Landing Pts:  400  Penalty Pts:  -20
   148       727        6      3 s     Runways : DRY
                               3 e     Wind    : 0 - 20 knots from SOUTH
-> 692       747        4      3 w
                               2 n          Flts in Queue: ......
                               2 s               <F1> to accept
   428      prop      * 3      2 e
                               2 w
   259       727        4      1 n
                               1 s
   840      prop        4      1 e
   190      DC10        5      1 w

n =========================  s  #1

n ==================         s  #2

w |||||||||||||||||||||||||  e  #3

w ||||||||||||||||||         e  #4
```

FIG. 3. A prototypical display of the Kanfer–Ackerman ATC task.

changes, one for providing feedback on errors, and one for displaying the rules of the task in response to the requests made by the participants. The 12 hold-pattern positions are divided into three levels corresponding to altitude, with hold-level 3 being the highest and hold-level 1 being the lowest.

Participants must observe the following set of rules while performing this task.

Rule 1: They must land planes into the wind. For instance, if the wind is from the north, they must land the plane on a north/south runway.

Rule 2: They can only land a plane from the first hold-level.

Rule 3: They must only move a plane one hold-level at a time, and they must only move a plane into an open hold-position.

Rule 4: While they can always land a plane on a long runway, they can only use a short runway when the conditions of the weather and the runways permit. In particular, they can only land a DC10 on a short runway when the runways are not ICY and the wind speed is less than 40 knots. In addition, they can only land a 727 on a short runway when the runways are DRY or the wind speed is 0–20 knots. And, regardless of the conditions of the runways and the weather, they can never land a 747 on a short runway, while PROP's, on the other hand, can always be landed on a short runway.

Rule 5: They can only land one plane at a time on a runway.

Rule 6: They are penalized for each minute that a plane dips below 4 min of fuel.

Participants can execute three actions in this task as follows: (a) they can accept a plane from the queue into an open hold-position, (b) they can move a plane between the three hold-levels, and (c) they can land a plane on a runway. They can accomplish these actions by using four keys: the Up-arrow and the Down-arrow keys, (↑ and ↓), the F1 function key (F1), and the Enter key (↵). They can move the cursor up and down the hold-positions and the runways using the Up- and Down-arrow keys. They can accept a plane from the queue into an open hold-position using the F1 key. And, they can select a plane in the hold, place a selected plane in an open hold-position (either from the queue or from another hold-position), or land a plane on a runway using the Enter key. In addition, participants can press the number keys 1–6 to examine Rules 1–6 any time during the task. They are given 50 points for landing a plane, penalized 100 points for crashing a plane, and penalized 10 points for violating one of the six rules. A plane crashes when the fuel level of a plane falls to 0 min.

Critical features of KA-ATC task are that planes are added to the queue approximately every 7 s and that it takes 15 s for a plane to clear a runway. Since only two runways can be used at a time (depending on wind direction) people can never exhaust the planes in the queue over the course of the experiment. However, planes in the queue also do not use up fuel so there is no real pressure or incentive to empty out the queue. But, once planes are entered from the queue into the hold-levels, they have between 4 and 6 min of fuel and begin to lose fuel in real time. Hence, they need to be attended to quickly to avoid a low-fuel penalty (i.e., Rule 6) or even crashing them when the fuel level goes to 0.

We should note that KA-ATC task is typically run under two conditions: full-weather and fair-weather conditions. In the full-weather condition, the task is run under the conditions as described previously. In the fair-weather condition, the wind speed is permanently set to 0–20 knots and the runway condition is permanently set to DRY. The main difference between the two conditions is that, in the full-weather condition, the cognitively complex part of this task concerns keeping track of what type of planes can be landed on the short runway under the different weather conditions (i.e., Rule 4). However, in the fair-weather condition, Rule 4 is logically reduced to the simple rule that all planes, except 747s, can land on the short runway.

In the next section, we describe our hierarchical decomposition of the KA-ATC task in detail. Our method of task analysis is based on Card, Moran, and Newell's (1983) unit-task analysis. We therefore begin by briefly describing their method of task analysis before continuing with our decomposition of the learning in the KA-ATC task.

THE TASK ANALYSIS

Card, Moran, and Newell (1983) proposed a method of task analysis in which a task is decomposed and analyzed at three, increasingly specific, levels.[2] The three levels are (a) the unit-task level, (b) the functional level, and (c) the keystroke level. The unit-task level is the most general level of analysis. At this level, the main task is decomposed into a set of independent unit-task goals that are repeatedly executed to achieve the main-task goal. The functional level is the level below the unit-task level in which the unit-task goals are further decomposed into even smaller, functional-level goals. The keystroke level is the most detailed level of analysis and consists of elementary motor and cognitive goals. The keystroke-level goals include goals to press keys, to find and encode information from the environment, and to retrieve information from long-term memory.

The central premise of Card et al.'s methodology for unit-task analysis is that a task can be decomposed into increasingly specific goals, all the way down to the keystroke level of elementary cognitive and perceptual-motor goals. Of course, they recognized that not all tasks can be easily decomposed this way, but they argued that for those tasks that can be decomposed, one could then analyze human performance at these different levels. While Card

[2] Card, Moran, and Newell (1983) also defined the argument level, which lies between the functional level and the keystroke level. The argument level consists of instantiations of the functional-level goals. For the purpose of our analysis and our argument, decomposition at the argument level was not needed and was therefore left out.

et al. focused primarily on skilled performance, the unit-task methodology can be readily extended to the analysis of *learning* at these different levels.

Figure 4 illustrates our decomposition of the KA-ATC task. The overall task can be decomposed into performing a sequence of three unit-tasks. The three unit-tasks are (a) moving a plane between the hold-levels, (b) landing a plane on a runway, and (c) getting a plane from the queue into a hold-position. Optimal performance in this task depends on the appropriate selection and parameterization of these unit-tasks. For instance, Reder and Schunn (1999) have shown that in the full-weather condition, better performance is obtained by people who use short runways whenever possible to maximize the availability of long runways for 747s and other planes that require them because of the unfavorable weather conditions. In the fair-weather condition, Lee, Anderson, and Matessa (1995) demonstrated that better performance is obtained by people who bring planes directly into hold-level 1 from the queue, thereby minimizing the number of keystrokes required to land a plane.

As Fig. 4 further illustrates, each unit-task can be decomposed into a number of functional-level goals. For instance, the unit-task of landing a plane involves (1) finding a plane to land, (2) moving to the plane, (3) selecting the plane, (4) finding a runway to land, (5) moving to the desired runway, and (6) landing the plane. Each of these functional-level goals involves a number of keystroke-level goals, including a sequence of shifts of attention across the screen, encoding of information on the screen, and a keystroke to effect the desired action. These functional-level goals fall into one of three categories: (1) finding an object, (2) moving to that object, and (3) selecting that object. To help identify these categories, we label the finding-an-object category with the *Find* prefix, the moving-to-an-object category with the *Move to* prefix, and the selecting-an-object category with the *Select* prefix. Tables 1a–1c define and classify the keystrokes for the land, move, and queue unit-tasks, respectively, according to the functions they serve.

Figure 5 illustrates the mapping between the functional-level goals and their associated latencies. As can be seen in Fig. 5, the total time to complete a unit-task is the sum of the total time to perform its functional-level goals. With the exception of the two *Move to* goals, the latency to complete a functional-level goal corresponds to a single keystroke time. As for the two *Move to* goals, to standardize our analysis of the functional-level goals to a single keystroke time, we aggregated the multiple arrow-key presses in the *Move to* goals into a single keystroke time by calculating and using their mean. While we associate the time to complete a functional-level goal with the time to complete a keystroke, this does not mean that these goals only contain the keystroke time. They also include the time for the required cognitive processing and visual information encoding that preceded the keypress. This will become clear when we discuss the data from our decompositional analysis of the latencies in the next section.

From our task analysis of the KA-ATC task, we develop two hypotheses.

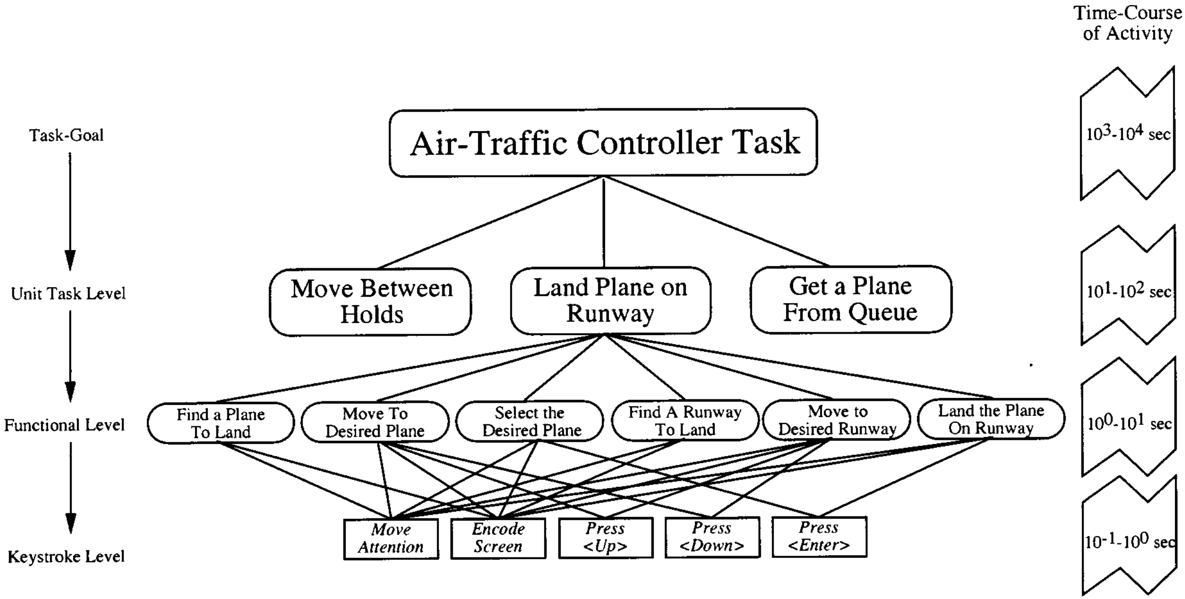


FIG. 4. A task analysis of the Kanfer–Ackerman ATC task.

TABLE 1a
Decomposition of the Land Unit-Task into Functional-Level Goals

Keystrokes	Definitions
Find Plane	Getting to the desired plane involves hitting a sequence of arrow keys until the cursor (which is at the left side of the hold) points to it. The time to press the first keystroke in this sequence is associated with finding the plane to land.
Move to Plane	The remaining arrow keys required to get to the desired plane are aggregated into this second catetory.
Select Plane	Selecting a plane involves hitting the Enter key, at which point the cursor switches to the right side of the hold-pattern. The time to press the Enter key is associated with the completion of this goal.
Find Runway	Getting to the desired runway involves hitting a sequence of arrow keys until the cursor (which is currently on the right side of the hold) points to it. The time to press the first keystroke in this sequence is associated with finding the runway on which to land.
Move to Runway	The remaining arrow keys required to get to the desired runway are aggregated into this category.
Select Runway	Selecting a runway involves hitting the Enter key, at which point the cursor returns to the hold-level and the plane will begin to taxi across the runway, taking 15 s to land. However, participants are free to begin the next unit-task while the plane is taxiing across the runway. The time to press the Enter key is associated with the completion of this goal.

TABLE 1b
Decomposition of the Move Unit-Task into Functional-Level Goals

Keystrokes	Definitions
Find Plane	Getting to the desired plane involves hitting a sequence of arrow keys until the cursor (which is at the left side of the hold) points to it. The time to press the first keystroke in this sequence is associated with finding the plane to move.
Move to Plane	The remaining arrow keys required to get to the desired plane are aggregated into this catetory.
Select Plane	Selecting a plane involves hitting the Enter key, at which point the cursor switches to the right of the hold-level. The time to press the Enter key is associated with the completion of this goal.
Find Hold	Getting to the desired hold-position involves hitting a sequence of arrow keys until the cursor (which is currently on the right side of the hold) points to it. The time to press the first keystroke in this sequence is associated with finding the desired hold-position.
Move to Hold	The remaining arrow keys required to get to the desired hold position are aggregated into this category.
Select Hold	Selecting a hold involves hitting the Enter key, at which point the plane will move to the new hold-level. The time to press the Enter key is associated with the completion of this goal.

TABLE 1c
Decomposition of the Queue Unit-Task into Functional-Level Goals

Keystrokes	Definitions
Find Hold	Selecting a hold into which to enter a queued plane involves hitting a sequence of arrow keys until the cursor (which is at the left of the hold) points to an empty hold-position. The time to press the first keystroke in this sequence is associated with finding the empty hold.
Move to Hold	The remaining arrow keystrokes required to move to the empty hold are aggregated into this category.
Select Queue	Getting a plane from the Queue involves hitting the F1 key to select the Queue. The time to press the F1 key is associated with the completion of this goal.
Select Hold	Selecting a hold involves pressing the Enter key to complete the queue unit-task. After the Enter key is pressed, a new plane appears in the selected hold position. The time to press the Enter key is associated with the completion of this goal.

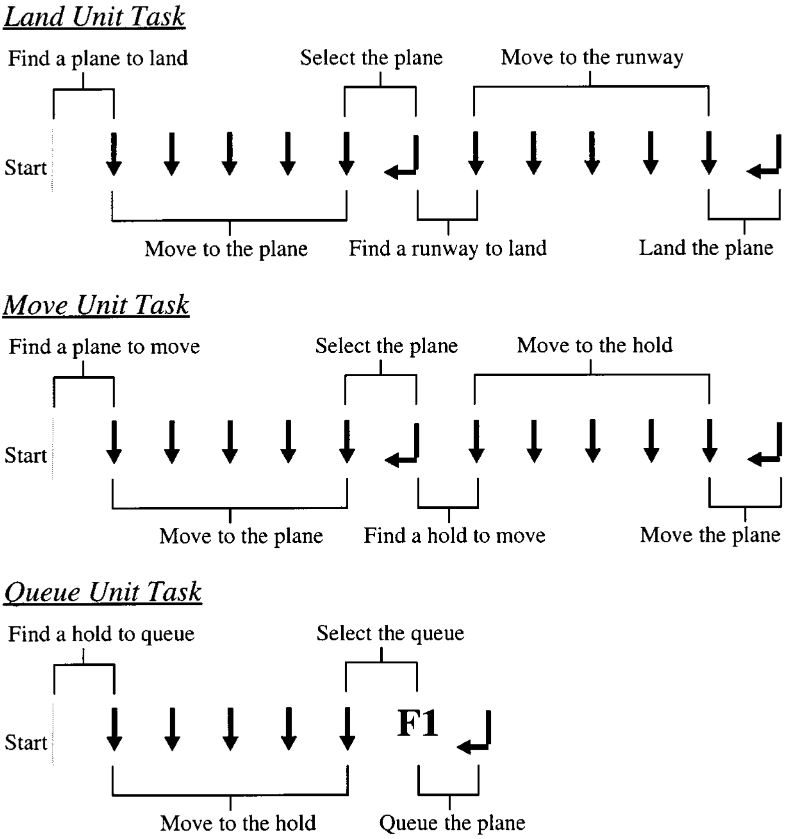


FIG. 5. The mapping between unit-tasks and their functional-level goals to keystrokes.

First, at the unit-task level, we hypothesize a rank order among the latencies of the unit-task goals. That is, on average a land unit-task will take longer than a move unit-task and a move unit-task will take longer than a queue unit-task. While it may be obvious that a queue unit-task should take the shortest amount of time, since it consists of fewer number of keystrokes, it is less obvious why a move unit-task should take less time than a land unit-task, since they are nearly identical in their keystrokes. However, the cognitive requirements of the two unit-tasks are quite different. A land unit-task requires making more complex decisions, i.e., whether a plane can be landed on a runway given the current weather condition, and accessing additional screen information, e.g., the current wind direction, than is required for a move unit-task. Second, at the functional level, we hypothesize a rank order among the latencies of the functional-level goals. That is, on average a goal for choosing an object (i.e., *Find*) and for selecting an object (i.e., *Select*) will take longer than a goal to move to an object (i.e., *Move to*). This is because both the *Find* and the *Select* goal require much more cognition than a *Move to* goal. That is, during a *Find* goal, people must decide which object they want to work on next. After finding an object, a *Move to* goal involves simply moving to that object. After moving to the object, they then must decide whether they want to select it (i.e., commit to it). These additional decisions required for the *Find* and *Select* functional-level goals will reflect in their longer latencies compared to a *Move to* goal. With these two hypotheses in mind, we now apply our decompositional analysis to the data from Study 2 in Ackerman and Kanfer (1994).

DATA FROM STUDY 2 OF ACKERMAN AND KANFER (1994)

Performance in the ATC task

The data from Study 2 in the ONR data set (Ackerman & Kanfer, 1994) come from Ackerman (1988). It is typical of the data in this set, and we present an analysis of only this one data set to be brief. It is also the data set that has been designated as the target modeling data set for the ONR Hybrid Architectures Project. The data from Study 2 were from 65 college undergraduates who completed 27 trials of the KA-ATC task with each trial lasting 10 min. The first 18 trials were in the fair-weather condition and the last 9 were in the full-weather condition. In our reanalysis of Ackerman (1988), we only examine the data from the fair-weather trials (i.e., the first 18 trials) of the 50 of the 65 participants who successfully completed all 27 trials.

Figure 6 presents an analysis of aggregate performance in terms of the mean time to land a plane and the mean number of keystrokes issued per landing. As can be seen, there is a dramatic improvement in performance over the course of 18 trials. Part of the overall improvement can be explained by selection of better strategies for landing planes and eliminating errors.

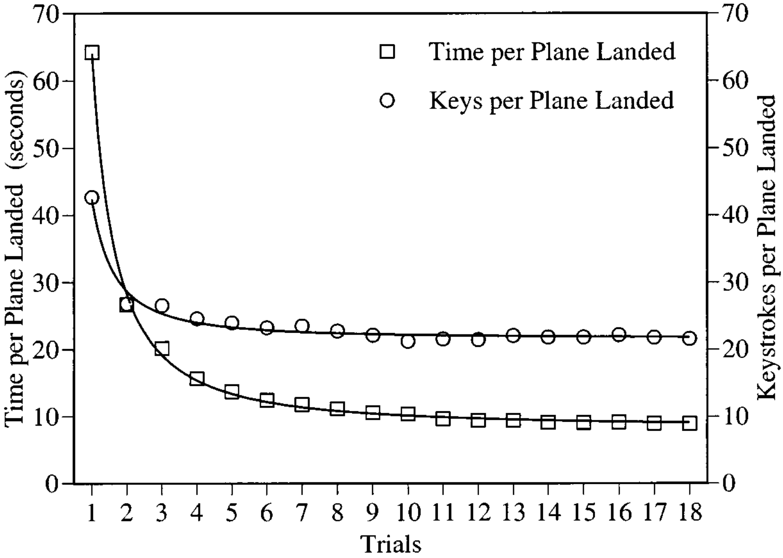


FIG. 6. The mean time to land one plane and the mean keystrokes used per landing from Ackerman (1988).

This part of the improvement is reflected in the decrease in the number of keystrokes per landing, which decreases from 42.7 keystrokes per landing on Trial 1 to 21.6 keystrokes per landing by Trial 18. However, people require 64.2 s to land a plane on Trial 1 but only 8.9 s to land a plane by Trial 18. Clearly, the improved performance in terms of strategy efficiency, as measured by the reduction in the number of keystrokes per landing, is not sufficient to account for the dramatic improvement in the overall task performance, as measured by the time to land a plane. People are improving by a factor of more than 7 in the time to land a plane but only decreasing in the number of keystrokes by a factor of less than 2.

Unit-Tasks

Figure 7 displays the critical factor underlying the improvement in the KA-ATC task, and this is the mean time to complete the three unit-tasks (note that for our analysis, we only include those unit-tasks that were successfully completed). As described previously in Fig. 5, the time to perform a unit-task is defined as the time from the last keystroke of the previous unit-task that caused a plane to land, move, or be accepted from the queue to the last keystroke that completed the current unit-task. As can be seen in Fig. 7, the latencies of the unit-tasks are speeding up by a factor of about 3 over the course of the experiment, and these improved unit-task performances are the major sources of the improvement in the overall task performance.

Figure 7 also plots the results of fitting power functions to the three unit-

Unit-Tasks	Power Functions	R^2	χ^2	S.E.
Land Unit-Task	$T = 3582 + 8091\ N^{-0.733}$	0.997	5.061	187
Move Unit-Task	$T = 2590 + 6144\ N^{-0.733}$	0.991	13.040	147
Queue Unit-Task	$T = 768 + 6215\ N^{-0.733}$	0.971	26.544	189

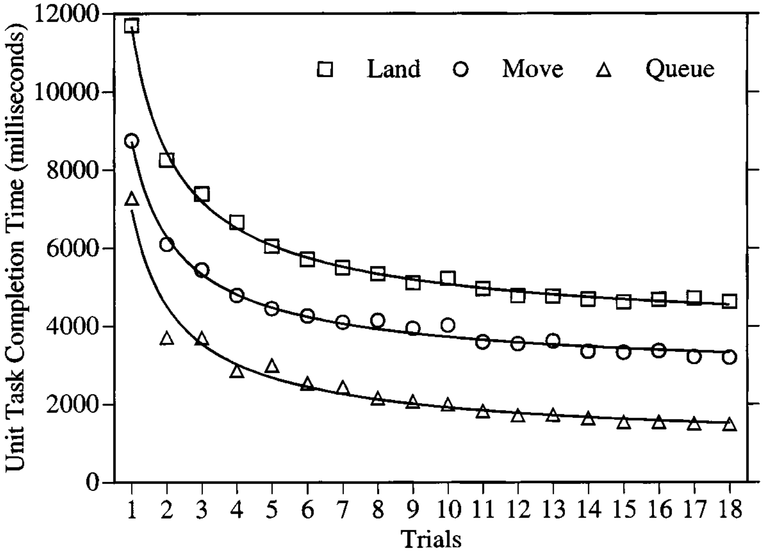


FIG. 7. The mean time to complete land, move, and queue unit-tasks and their power-function fits.

tasks. While we estimated different asymptotes and scale factors for the three unit-tasks, we constrained them to have a common exponent, 0.733, which was estimated. As can be seen, the fits are quite good even with this constraint. One can measure the consequences of the constraint by a chi-square measure of fit. We calculated the chi-square measure of fit by first obtaining a measure of the variance in the data by calculating for each curve the participant-by-trial interaction and then deriving the standard errors of measurement. The chi-square is the ratio of the sum of squared deviations and the square of these standard errors. Throughout this article we report power function fits obtained by minimizing the chi-square measures. The asymptotes 3582, 2590, and 768 ms in Fig. 7, can be interpreted as the minimal times to perform each of these unit-tasks, and the scale factors 8091, 6144, and 6215 ms, can be interpreted as the amount of time that can be compressed with practice.

Our decision to constrain the model to a single exponent was based on the goodness-of-fit provided by the model relative to its degrees of freedom (cf. Anderson, 1989). That is, if we estimate three separate exponents for the three curves, the total χ^2 is 42.759, whereas if we constrain the exponents to be the same, the total χ^2 is 44.645. The number of degrees of freedom for the chi-square of the constrained exponent model is the number of observations ($3 \times 18 = 54$) minus the 7 estimated parameters (3 asymptotes + 3 scale factors + 1 exponent) for a total of 47 degrees of freedom. As can be seen, the observed chi-square is approximately equal to its degrees of freedom, which is one sign of the goodness of the fit. Also, the difference between the two χ^2 is less than 2, and this difference corresponds to a χ^2 of 2. Thus, the improvement in the fit of the unconstrained model with the three separate exponents is not significantly better than the fit of the constrained model with a single exponent.

Note from Fig. 7 that the mean time to complete a land unit-task is longer than the mean time to complete a move unit-task, which in turn is longer than the mean time to complete a queue unit-task across the 18 trials. The greater scale factor for a land unit-task indicates that there are more cognitive components to be learned for this unit-task, and its larger asymptote indicates that even at a skilled level of performance, it requires more information processing. The rank order of the latencies of the data are consistent with our hypotheses from our task analysis.

In this article, we plot the learning data as a function of time (i.e., trials) rather than as a function of the number of times practiced. This raises the issue of whether the data or the power functions would have been very different had we plotted the data as a function of practice. To address this issue, we replotted the unit-task data from Fig. 7 as a function of practice in Fig. 8, with each block representing the mean of 40 unit-tasks. Figure 8 also plots the best fitting power functions to the replotted data with a common exponent of 0.844, which was estimated. As can be seen in Fig. 8, the power functions provide excellent fits to the replotted data. Newell and Rosenbloom (1981) showed that a power function in terms of time (Fig. 7) implied a power function in terms of opportunities (Fig. 8) or vice versa. In addition, the replotted data are completely consistent with the data from Fig. 7, namely the rank order of the unit-task execution times between the two plots are identical. Since our only concern is that the unit-tasks are being learned according to a well-defined learning function, and since a power function of one form implies the other, we do not plot further the functions both ways. We plot the data as a function of time (Fig. 7), since this is the more convenient aggregation.

Functional-Level Goals

We next consider the latencies for the individual keystrokes that correspond to the different functional-level goals (see Figs. 4 and 5). The data are plotted in Fig. 9a for the functional-level goals making up the land unit-task, Fig. 9b for the functional-level goals making up the move unit-task, and Fig. 9c for the functional-level goals making up the queue unit-task. These data give the appearance of relatively continuous learning curves. To

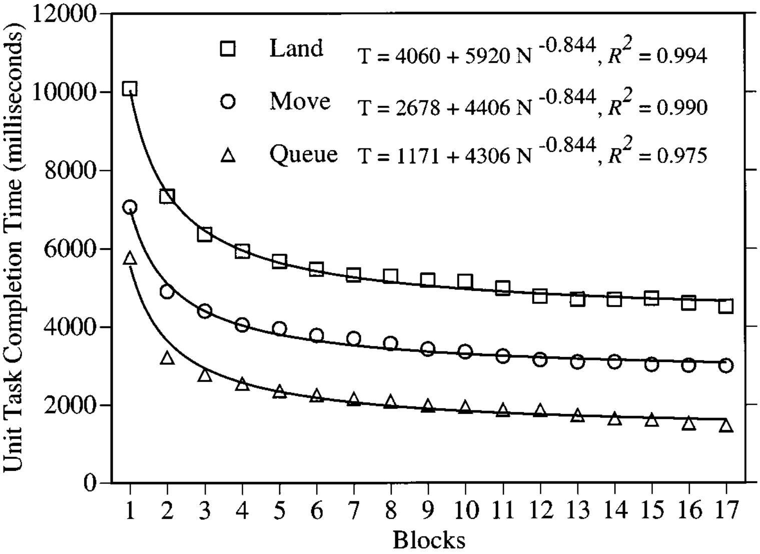
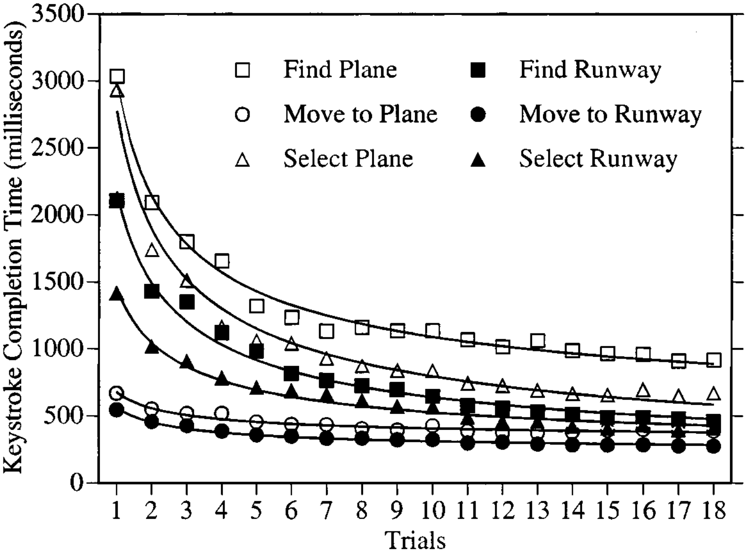


FIG. 8. The mean time to complete land, move, and queue unit-tasks as a function of the number of unit-tasks performed. Each block represents the mean of 40 unit-tasks performed.

explore the issue of the shape of these learning curves, we tried fitting a variety of power functions. In fitting the power functions, we considered the possibility of constraining the various parameters to be the same for all the learning curves. We examined constraining all possible combinations of the three power-function parameters (A—Asymptote, S—Scale Factor, and E—Exponent), which gave us eight possible models (letters indicating the parameters that were constrained): {}, {A}, {S}, {E}, {AS}, {AE}, {SE}, and {ASE}, from the completely unconstrained model, {}, to the completely constrained model, {ASE}. Our decision in choosing a power-function model was based on our analysis of this space of the constrained power-function models as given in Fig. 10. The numbers in each cell of Fig. 10 represent the R_2 goodness-of-fit and the chi-square measure of deviation for that model. Most critical result from this analysis is that the unconstrained model, {}, with a total χ^2 of 241 (df = 240) does not significantly deviate from the data. This implies that all components correspond to continuous learning functions. In addition, we lose relatively little in terms of R^2 between the unconstrained model, {}, and the two single-parameter constrained models, {A} and {E}, or the double-constrained, {AE}, model. This implies that these functions have approximately the same asymptotes and exponents. The double-constrained {AE} model had a best fitting asymptote of 198 ms and a best fitting exponent of 0.531. Note the value of the common asymptote is about the average time to type a character. While the {A} model provides a slightly better fit than the {E} model for this data, the {E} model provides

Unit-Tasks	Power Functions	R^2	χ^2	*S.E.*
Find Plane	$T = 373 + 2609\ N^{-0.561}$	0.988	9.178	79
Move to Plane	$T = 304 + 375\ N^{-0.561}$	0.954	12.786	20
Select Plane	$T = 50 + 2720\ N^{-0.561}$	0.983	20.105	69
Find Runway	$T = 61 + 2111\ N^{-0.561}$	0.985	15.453	56
Move to Runway	$T = 216 + 349\ N^{-0.561}$	0.978	16.919	11
Select Runway	$T = 175 + 1283\ N^{-0.561}$	0.982	47.047	22
**Select Runway*	$T = 50 + 1378\ N^{-0.457}$	*0.982*	*29.546*	22



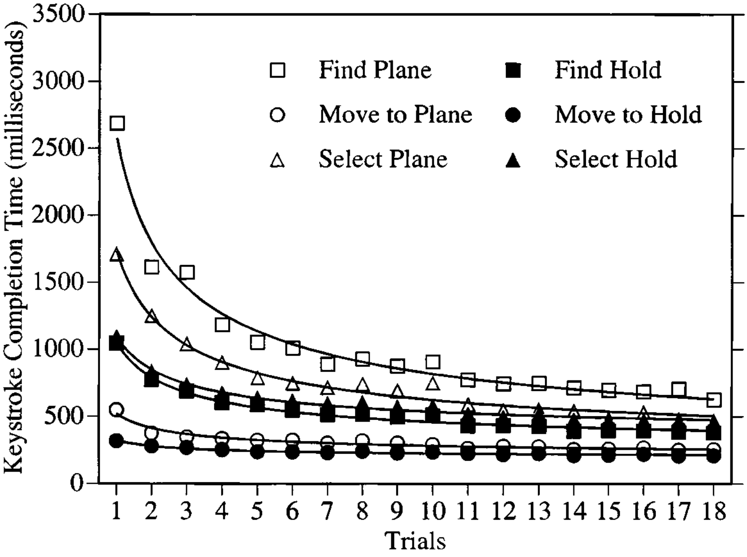
* We have estimated separate exponents for those curves that showed large deviations from the common exponent and have provided them here (in italics) in addition to the original constrained functions.

(a)

FIG. 9. (a) The mean time to complete the keystrokes associated with the functional-level goals of the land unit-task, (b) the mean time to complete the keystrokes associated with the functional-level goals of the move unit-task, and (c) the mean time to complete the keystrokes associated with the functional-level goals of the queue unit-task.

a superior fit for all the other data that we examine in this article. Also, constraining the exponent means that all the curves have all the same shape and allows us to compare the other two parameters across curves, with S reflecting the amount to be learned and A reflecting minimum time. Hence, we chose to present the {E} model in Fig. 10 with the exponent constrained

Unit-Tasks	Power Functions	R^2	χ^2	*S.E.*
Find Plane	$T = 153 + 2421\, N^{-0.561}$	0.979	13.197	83
Move to Plane	$T = 190 + 330\, N^{-0.561}$	0.944	30.663	12
**Move to Plane*	$T = 235 + 302\, N^{-0.821}$	*0.944*	*25.785*	*12*
Select Plane	$T = 203 + 1530\, N^{-0.561}$	0.982	21.262	38
Find Hold	$T = 235 + 891\, N^{-0.561}$	0.984	16.543	22
Move to Hold	$T = 188 + 134\, N^{-0.561}$	0.958	16.570	6
Select Hold	$T = 322 + 777\, N^{-0.561}$	0.983	20.093	19



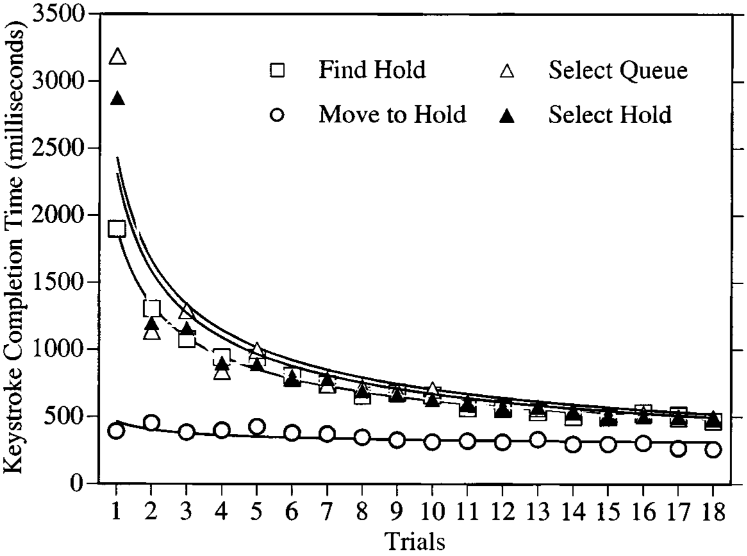
* We have estimated separate exponents for those curves that showed large deviations from the common exponent and have provided them here (in italics) in addition to the original constrained functions.

(b)

FIG. 9—*Continued*

and estimated to 0.561. As an additional note, when we estimate the power functions for the functional-level goals, we constrain the asymptote to be no less than 50 ms, based on the view that minimal motor actions have a fixed cost associated with them (Card, Moran, & Newell, 1983). In the following subsections, we analyze the functional-level goals for each unit-task in detail. These goals are fit with a common exponent to facilitate the comparison of other parameters. However, where there are significant deviations, we also report the unconstrained fits. We have chosen a chi-square of 30 as the

Unit-Tasks	Power Functions	R^2	χ^2	*S.E.*
Find Hold	$T = 138 + 1763\ N^{-0.561}$	0.992	6.207	53
Move to Hold	$T = 275 + 192\ N^{-0.561}$	0.528	63.264	19
*Move to Hold	$T = 50 + 409\ N^{-0.169}$	*0.672*	*44.117*	*19*
Select Queue	$T = 50 + 2384\ N^{-0.561}$	0.879	39.929	159
*Select Queue	$T = 534 + 2600\ N^{-1.461}$	*0.960*	*10.553*	*159*
Select Hold	$T = 50 + 2261\ N^{-0.561}$	0.915	33.364	128
*Select Hold	$T = 469 + 2334\ N^{-1.300}$	*0.978*	*7.160*	*128*



* We have estimated separate exponents for those curves that showed large deviations from the common exponent and have provided them here (in italics) in addition to the original constrained functions.

(c)

FIG. 9—*Continued*

threshold for significance in these analyses. This corresponds approximately to a significance level of $\alpha = 0.01$. With so many curves, we selected a relatively high threshold for significance to avoid Type I error.

Functional-level goals of the land unit-task. One can examine the latencies for the functional-level goals of the land unit-task by examining their associated keystroke completion times. Figure 9a plots the keystroke completion times associated with the six functional-level goals of the land unit-task and their fitted power functions. As we have indicated previously, in fitting the

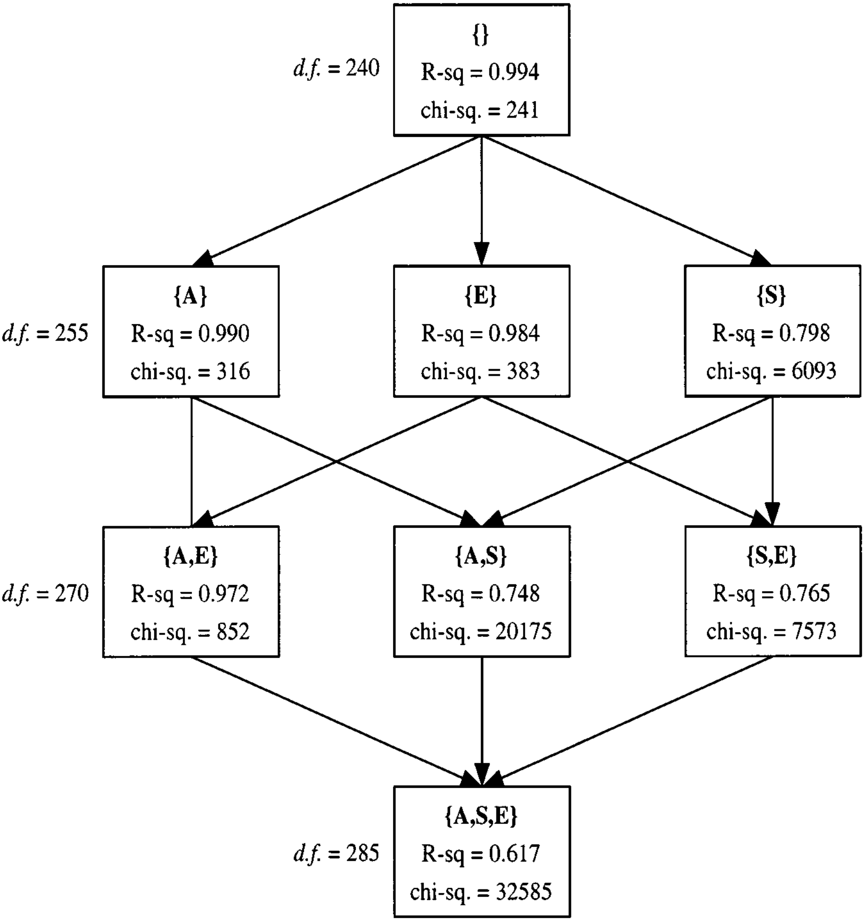


FIG. 10. A lattice representing the total combination of power-function models.

power functions to these six individual curves, we constrained their exponent to be 0.561. It is clear from Fig. 9a that the times for finding and selecting planes and finding and selecting runways are much longer than moving to those planes and runways. That is, the large latency and the scale-factor differences between the *Find* and *Select* goals relative to a *Move to* goal show that not all keystrokes speed up uniformly. This is consistent with our expectations from our task analysis that the long keystrokes are those associated with the goal of identifying targets (i.e., the *Find* goal) and recognizing when the goal has been achieved (i.e., the *Select* goal). We see this same pattern of latencies in the functional-level goals of other unit-tasks as well.

Except for *Select Runway,* the functional-level goals of the land unit-task are well fit by power functions with the common exponent of 0.561. Therefore we estimated a separate exponent for *Select Runway,* which turned out to be 0.457, and the χ^2 dropped to 29.546. The χ^2 measure of fit for the rest of the functional-level goals is 74.440 with 80 degrees of freedom, which is not significant. The asymptotes for all the power functions suggest that the minimum time is coming close to the minimum keystroke time.[3]

Select Runway is particularly complicated by the fact that on some landings people will get a plane to the runway only to have to wait for it to clear. To eliminate this waiting time from the measurement of the *Select Runway,* we took the lesser of (1) the time to press the Enter key from the previous arrow key and (2) the time to press the Enter key from the time that the chosen runway clears. The quality of the fit is much worse if we do not do this. The fact that the fit for the *Select Runway* is still significantly deviant suggests that we have not quite derived the right measure of processing time for it. Currently, we slightly overpredict its latency in the final trials.

Functional-level goals of the move unit-task. One can examine the times for finding, moving to, and selecting planes and hold-positions by examining their keystroke completion times. Figure 9b plots these times plus their best fitting power functions. In fitting the power functions, the exponent was constrained to be 0.561. As can be seen, all the power functions provide good fits with the exception of *Move to Plane*. Therefore, we estimated a separated exponent for *Move to Plane,* 0.821, which lowered the χ^2 from 30.663 to 25.785. The overall χ^2 measure of fit is 118.329, with 96 degrees of freedom, which is not significant. Again all of the asymptotes are close to a single-keystroke time. As can be seen in Fig. 9b the latencies for finding and selecting planes and finding and selecting hold-positions are much longer than the latencies to move to the planes and the hold-positions. The differences in the latencies for the *Find* and the *Select* goals versus the *Move to* goals are consistent with our expectations from the task analysis.

Functional-level goals of the queue unit-task. The queue unit-task involves one less *Find* and *Move to* to goals compared to the land and the move unit-tasks. In addition, the queue unit-task has a function-level goal, *Select Queue,* which is unique to it and involves hitting the F1 key. Figure 9c plots the latencies associated with the functional-level goals of the queue unit-task, and their best-fitting power functions. As can be seen in Figure 9c, the latencies for the *Find Hold, Select Queue,* and *Select Hold* are all approximately the same and speed up at approximately the same rate. The overall χ^2 measure of fit for the power functions is 142.763 with 64 degrees of freedom,

[3] Card, Moran, and Newell (1983) estimated the minimal keystroke time to be about 70 ± 30 ms for a keystroke that only contained a simple downstroke to press the key and twice that time, i.e., 140 ± 60 ms, for a keystroke that contained both the downstroke to press the key and upstroke to return the finger to its original position. We therefore take the minimal keystroke time to range anywhere from 40 to 200 ms.

which is significant. None of the power functions, with the exception of the one for *Find Hold*, provide a good chi-square fit to the data. We therefore estimated separate exponents for the *Move to Hold, Select Queue*, and *Select Hold*. The estimated exponents were 0.169, 1.461, and 1.300, respectively. Of these, only *Move to Hold* remained significantly deviant from the data. The most likely reason for the poor fit of the *Move to Hold* is that people are already at the asymptote from Trial 1 for this keystroke, and the small variations from trial to trial for *Move to Hold* are mostly due to noise. Regardless, it is clear that the latencies for the *Find* and *Select* goals are much longer than the latencies for the *Move to* goal, reflecting the higher cognitive costs associated with them. Additionally, the asymptotes of the power functions are in the range of a single keystroke time.

Summary

The overall improvement in the KA-ATC task reflects an improvement in the strategies for landing planes, as indexed by the reduction in number of keystrokes per landing (Fig. 6), and an improvement in the execution of unit-tasks and their functional-level goals, as reflected in a reduction in the time per keystroke (Fig. 9). Both factors show power-law speed-up, but we have focused on the time per keystroke, as it was the more important factor. As can be seen in Fig. 9, with few exceptions, the constrained power function fits are quite good. The fact that some of the chi-squares are significant says more about the small standard errors than the fits themselves. The good fits overall support our argument that underlying a complex power function are many lower-level power functions. The fact that we were able to fit a single exponent to all the functional-level goals of the unit tasks, with reasonable success, suggests that they are speeding up at the same rate. In addition, the fact that their asymptotes are all in the range of a single-keystroke time indicates that people are speeding up to the minimum keying time. The differences among the learning curves largely reside in the scale factors that reflect their varying cognitive complexity. The clear implication of these learning curves is that the cognitive components are being squeezed down to zero. It still remains for us to identify more clearly what exactly is involved in these cognitive components. We address this issue in the following eye-tracking experiment. As an interim summary, we conclude that our reducibility thesis is supported by the fact that the learning at the different levels of decomposition is well fit by a power function. This is the *consistent learning* that we identified as the critical test of the reducibility hypothesis.

THE ATTENTIONAL SOURCE FOR SPEED UP IN PERFORMANCE

Implicit or explicit in many theories of skill acquisition is the view that the speed-up results from optimization of mental processes. For example, the prototypical view of the speed-up in learning the multiplication table is that it results from a strategy shift, i.e., shift from a calculate strategy to a memory retrieval strategy (Ashcraft, 1992; Lemaire & Siegler, 1995; Logan; 1988; Reder & Ritter, 1992; Rickard & Bourne, 1996; Siegler, 1988). Recently, however, Haider and Frensch (1996, 1999) have proposed another potential source of speed-up in learning. They hypothesized that some portion of the observed speed-up may actually result from the reduction of the processing of task-irrelevant information.

Haider and Frensch supported their hypothesis with results from an alphabet verification task in which people verified the correctness of an alphanumeric string that contained a varying number of task-irrelevant characters. In this task, as people became skilled, they increasingly became insensitive to the varying amount of task-irrelevant characters. Therefore Haider and Frensch argued that people learned to focus their information processing on the task-relevant subset of the string and increasingly reduced the information processing of the task-irrelevant subset of the string (Haider & Frensch, 1996). In a follow up eye-tracking experiment, they obtained direct evidence of the changes in people's attentional strategies (Haider & Frensch, 1999).

Epelboim et al. (Epelboim, Kowler, Edwards, Collewijn, Erkelens, & Steinman, 1994, as reported in Kowler, 1995) also found qualitative changes in attentional strategies with practice. In their task, people were exposed to a large breadboard, with 2, 4, or 6 vertical rods placed randomly, for 10 consecutive trials. In one condition, people were instructed to simply foveate on the pegs in a fixed color sequence, and in the other condition, they were instructed to foveate and tap with their finger on the pegs in a fixed color sequence. In both conditions, there was a gradual reduction of foveations to the irrelevant pegs in the sequence. As Epelboim et al. reported, ‘‘Gaze hops about all over the place during the first two or three repetitions while participant is searching for and learning the location of each rod. Things settle down by the fourth repetition.’’

With the exception of Haider and Frensch (1999) and Epelboim et al. (1994), there are relatively few other examples of experiments where researchers have looked at changes in eye movements of people learning to do a task. However, there are many examples of experiments where researchers have looked at the eye movements of experts versus novices in performing a task. Findings from these cross-sectional studies generally support the view that experts and novices attend to visual information differently. These tasks span a wide spectrum in complexity, and they include reading topological maps (Antes, Chan, Lenzen, & Mullis, 1985), putting in golf (Vickers, 1992), performing a dynamic videogame-like task (Shapiro & Raymond, 1989), managing a simulated electric bulb factory (Krappman, 1995), and analyzing X-ray films (Myles-Worsley, Johnston, & Simons, 1988).

Haider and Frensch (1996, 1999) used a relatively simple task. We were intrigued by the issue of how much of the improvement in the more complex KA-ATC task might also be due to decreased perceptual processing of task-irrelevant information. The screen of the KA-ATC task (Fig. 3) is clearly quite complex and offers a lot of visual information to distract people from their assigned task. It is possible that the time that was being eliminated in the functional-level goals in Fig. 9 was the time to search for relevant information, leaving only the keystroke time. Therefore, we ran an experiment in which we monitored the eye movements of people performing the KA-ATC task to see whether the learning in this task could also be explained by the attentional learning hypothesis forwarded by Haider and Frensch.

Method

Participants

Ten people from the Carnegie Mellon University community were recruited to participate in the eye-tracking experiment with the KA-ATC task. They were recruited from a database that we maintain internally of those who have previously participated in eye-tracking experiments. Each person performed 6 trials per day for 3 days, with each trial lasting 10 min, for a total of 18 trials in the fair-weather condition. While the time-on-task was 1 h for each day, we set aside an additional hour for such activities as equipment calibrations, task instruction, debriefing on the first and last days of the experiment, and for breaks between trials. People were paid $10.00/h, at the completion of their participation in the experiment.

Materials

The KA-ATC task, which was originally written for the PC/DOS platform, was ported over to the Macintosh platform using the Macintosh Common Lisp programming language. This was done in order for the task to work with our eye-tracking equipment and software. The eye-tracking equipment consisted of an ETL-500 video-based, head-mounted eye-tracking system with a magnetic-based head tracker from ISCAN, Inc. The software for collecting and analyzing the eye data consisted of the EPAL (Douglass, 1998) software suite that was internally developed in our lab to facilitate the development of eye-tracking experiments and the analysis of their data. The KA-ATC task was presented on an Apple 20-in. multisync monitor with its display resolution set to 1024 × 768 pixels. The monitor was placed approximately 3 ft from the participants, level with their eyes. The KA-ATC task window on the display monitor was set to 900 × 700 pixels, approximately 23° × 18° of visual angle.

Procedure

On the 1st day of the experiment, people worked through the interactive computer-based instructions for the KA-ATC task. The instructions consisted of a declarative component, in which they were introduced to the visual elements of the task, e.g., hold, runways, and queue, and the six rules that govern the task; and a procedural component, in which they were led through an example of the three unit-tasks, i.e., how to land a plane, how to move a plane between the hold-levels, and how to get a plane from the queue. The reading of the instructions was self-paced, and it typically lasted about 20 min. Eye movements were not recorded while participants read the instructions. After the instructions were read, participants were fitted with the eye-tracker and calibrated, a process that usually took about 5–10 min. Once they were calibrated, they began the first of six trials for that day. After each trial, their calibration was checked and recalibrated as needed. After each trial, they were given a choice to take a short two-min break. The same procedure was followed for the remaining days, as outlined for the 1st day, with the exception of the task instructions.

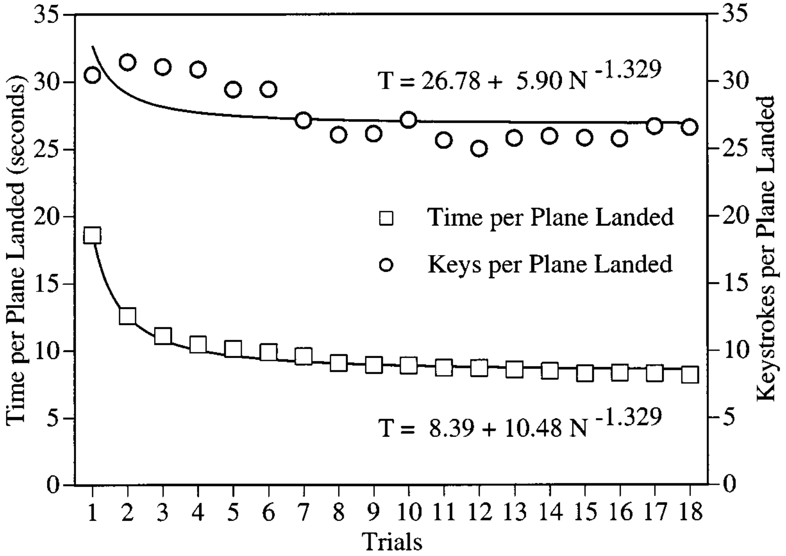


FIG. 11. The mean time to land one plane and the mean keystrokes used per landing.

Results

Before reporting the eye movement data, we report the more gross behavioral measures to establish the comparability of our data to those of Study 2 from Ackerman and Kanfer (1994). Figure 11 plots the mean time to land one plane per trial and the mean number of keystrokes per plane landed. It can be compared to Fig. 6. As can be seen in Fig. 11, participants require 18.6 s to land a plane at Trial 1 but only 8.2 s at Trial 18. However, keystrokes per plane landed change only slightly from 30.5 keystrokes at Trial 1 to 26.6 keystrokes at Trial 18. The improvement in the time to land a plane is not as dramatic as the improvement in the Ackerman data because our eye-tracking participants are a great deal faster from the start. However, they did improve by a factor of more than 2, and clearly, this improvement is not due to the reduction of the number of keystrokes per landing, which was negligible across the 18 trials.

Unit-Tasks

Fig. 12 plots the mean time to execute the three unit-tasks and their best fitting power functions. It can be compared to Fig. 7. As can be seen in Fig. 12, the latencies of the unit-tasks are speeding up by a factor of about 2 over the course of the experiment. If we estimate three separate exponents for the three curves, the total χ^2 is 29.630, and if we constrain the exponent to be the same, the total χ^2 is 36.283, not significant with 47 degrees of freedom. Thus, we estimated a single best fitting exponent for all three unit-tasks,

Unit-Tasks	Power Functions	R^2	χ^2	*S.E.*
Land Unit-Task	$T = 3035 + 3086\ N^{-1.329}$	0.963	6.344	232
Move Unit-Task	$T = 2627 + 2423\ N^{-1.329}$	0.956	11.483	148
Queue Unit-Task	$T = 1433 + 1911\ N^{-1.329}$	0.953	18.500	95

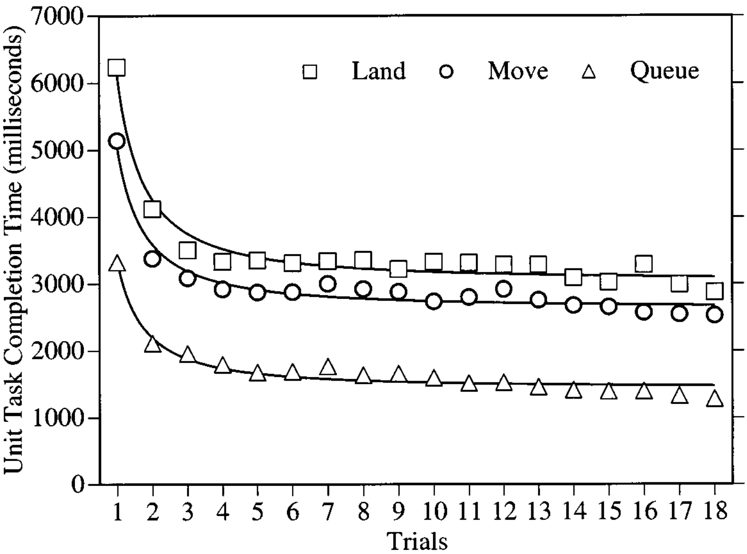
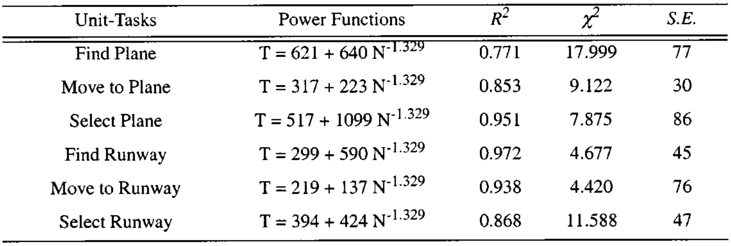


FIG. 12. The mean time to complete land, move, and queue unit-tasks and their power-function fits.

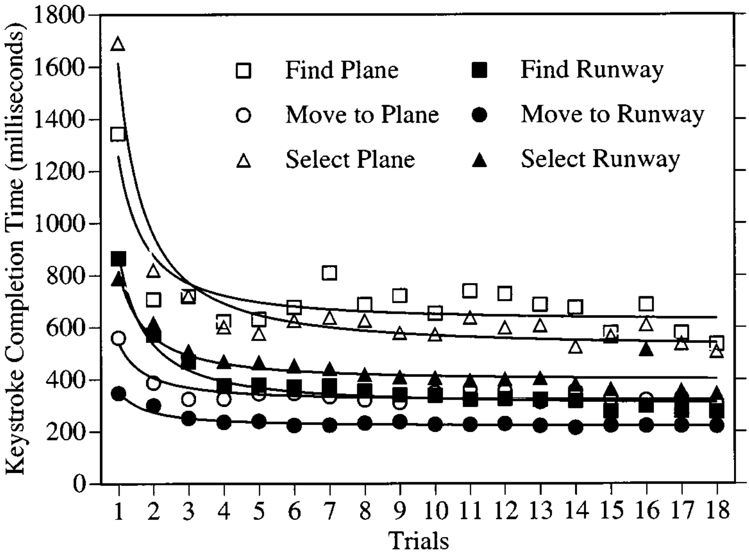
which turned out to be 1.329, and estimated separate asymptotes and scale factors for the three curves. The asymptotes (3035, 2627, and 1433 ms) can be interpreted as the minimal times to perform each of these unit-tasks, and the scale factors (3085, 2423, and 1911 ms) can be interpreted as the amount of time that can be compressed with practice. The mean time to complete a land unit-task is longer than the mean time to complete a move unit-task, which in turn is longer than the mean time to complete a queue unit-task, across the 18 trials. This is consistent with the expectations from our task analysis. While our participants are much faster than Ackerman's, the rank-order of the unit-task latencies of our data are identical to those of the Ackerman data.

Functional-Level Goals

Fig. 13 displays the latencies for the keystrokes corresponding to the functional-level goals of the land, move, and the queue unit-tasks. As we did for Ackerman's data, we examined the total space of the constrained power-function models in selecting a power function model for our data. Our analysis of the model space is given in Fig. 14 (the numbers in each cell represent



Unit-Tasks	Power Functions	R^2	χ^2	*S.E.*
Find Plane	$T = 621 + 640\,N^{-1.329}$	0.771	17.999	77
Move to Plane	$T = 317 + 223\,N^{-1.329}$	0.853	9.122	30
Select Plane	$T = 517 + 1099\,N^{-1.329}$	0.951	7.875	86
Find Runway	$T = 299 + 590\,N^{-1.329}$	0.972	4.677	45
Move to Runway	$T = 219 + 137\,N^{-1.329}$	0.938	4.420	76
Select Runway	$T = 394 + 424\,N^{-1.329}$	0.868	11.588	47



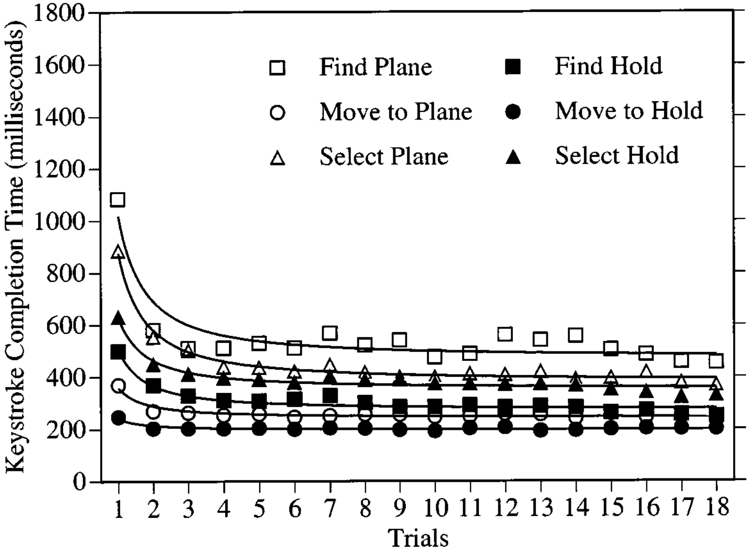
(a)

FIG. 13. (a) The mean time to complete the keystrokes associated with the functional-level goals of the land unit-task, (b) the mean time to complete the keystrokes associated with the functional-level goals of the move unit-task, and (c) the mean time to complete the keystrokes associated with the functional-level goals of the queue unit-task.

the R^2 goodness-of-fit and the chi-square measure of deviation for that model). As can be seen, we lose relatively little in terms of R^2 between the unconstrained model, {}, and the single-parameter constrained model {E}. Again, the most important to note here is that the unconstrained model, {}, with a total χ^2 of 126 (df = 240), does not significantly deviate from the data. In addition, the model with a constrained exponent does not significantly deviate from the data, total χ^2 of 180, df = 255. When we constrained both the asymptote and the exponent, {AE}, it resulted in a best fitting asymptote of 180 ms and a best fitting exponent of 0.413. Note that the best fitting asymptote for this model is close to the minimal keystroke time.

The best fitting exponent for the exponent-constrained {E} model was 1.531. We decided to explore the possibility of constraining the exponent further by setting the best fitting exponent for the functional-level goals to

Unit-Tasks	Power Functions	R^2	χ^2	*S.E.*
Find Plane	$T = 473 + 547\ N^{-1.329}$	0.862	15.898	53
Move to Plane	$T = 243 + 115\ N^{-1.329}$	0.911	7.204	13
Select Plane	$T = 382 + 494\ N^{-1.329}$	0.982	5.184	29
Find Hold	$T = 274 + 231\ N^{-1.329}$	0.927	17.660	15
Move to Hold	$T = 197 + 44\ N^{-1.329}$	0.797	15.503	5
Select Hold	$T = 354 + 275\ N^{-1.329}$	0.937	11.432	20



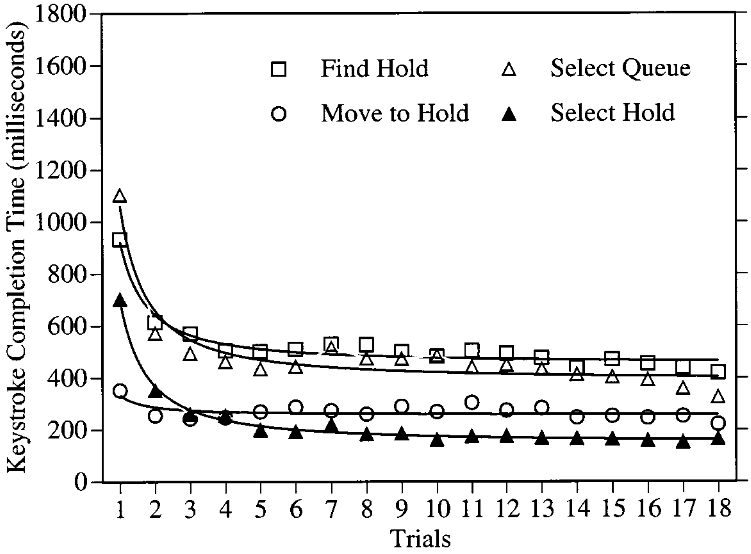
(b)

FIG. 13—*Continued*

be the same as the best fitting exponent from the unit-tasks, 1.329. When we did, we found a minimal change in the total χ^2 from 179.774 with {E = 1.531} and 186.117 with {E = 1.329}. Hence, we decided to constrain the exponent to the one estimated for the unit-tasks. Hence, the exponents of the power function fits to the functional-level goals have been constrained to 1.329.[4] Note, as before, when we estimate the power functions for the

[4] The reason why we used a common exponent for both the unit-tasks and the functional-level goals for our data, but not for the Ackerman data, was because the number of keystrokes per plane landed for our participants (Fig. 11) did not change very much across the 18 trials, whereas for Ackerman's participants, there was a dramatic drop. While the speed-up in the unit-tasks for our data was mostly due to the speed-up in the keystrokes, for the Ackerman data, the speed-up in the unit-tasks was also due to people using less keys to complete unit-tasks. Hence, we believed that it made sense to use a common exponent for both the unit-tasks and the functional-level goals for our data, since the speed-up in the unit-tasks mostly resulted from the speed-up of the individual keystrokes, whereas for the Ackerman data, it did not.

Unit-Tasks	Power Functions	R^2	χ^2	*S.E.*
Find Hold	$T = 455 + 469\ N^{-1.329}$	0.950	7.229	39
Move to Hold	$T = 257 + 76\ N^{-1.329}$	0.375	25.930	19
Select Queue	$T = 390 + 672\ N^{-1.329}$	0.917	16.288	49
Select Hold	$T = 150 + 547\ N^{-1.329}$	0.992	3.158	28



(c)

FIG. 13—*Continued*

functional-level goals, we constrain the asymptotes to be no less than 50 ms. In the following subsections, we analyze the functional-level goals for each unit-task in detail.

Functional-level goals of the land unit-task. Fig. 13a plots the mean latencies of the functional-level goals of the land unit-task and their best fitting power functions. They can be compared to those in Fig. 9a. The overall χ^2 measure of fit is 93.18, which is not significant with 96 degrees of freedom. As can be seen, the scale factors estimated for our data are much smaller compared to those estimated for Ackerman's data, indicating that our participants start out much faster at executing these goals. However, the relative differences in the latencies among the goals, i.e., that finding and selecting planes and runways takes longer than moving to those planes and runways, are identical to the Ackerman data and are consistent with our hypothesis from the task analysis.

Functional-level goals of the move unit-task. Fig. 13b plots the mean latencies of the functional-level goals of the move unit-task and their best fitting power functions. They can be compared to those in Fig. 9b. The overall χ^2

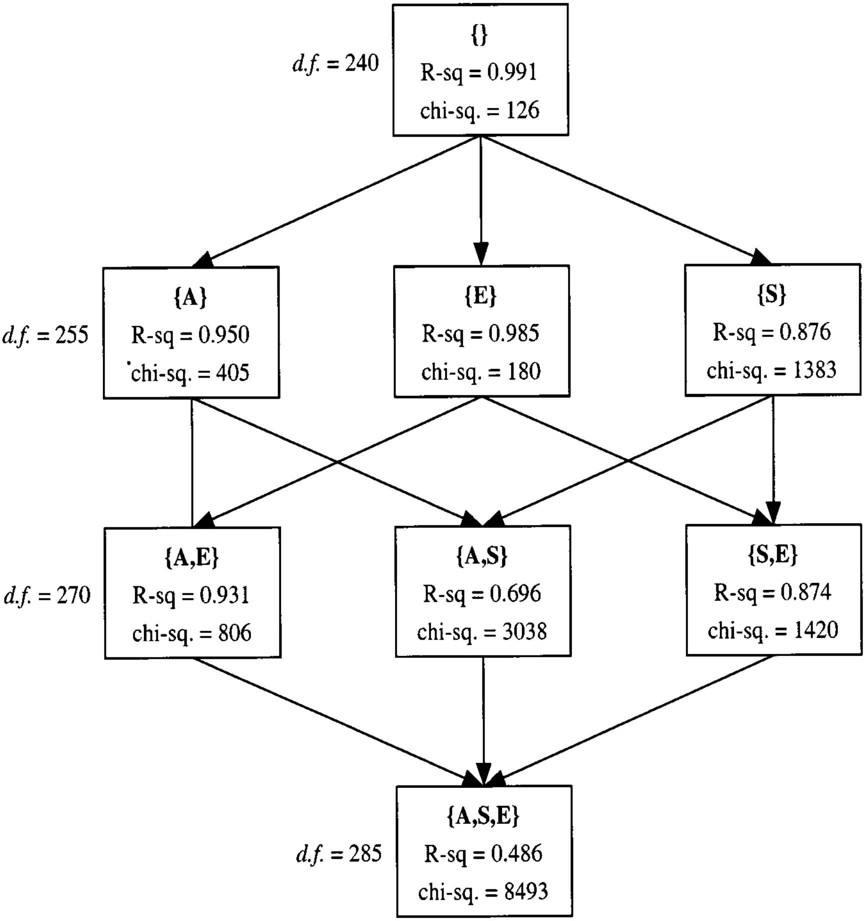


FIG. 14. A lattice representing the total combination of power-function models.

measure of fit is 87.41, which is not significant with 96 degrees of freedom. Again, the scale factors estimated for our data are much smaller compared to those estimated for the Ackerman data, indicating that our participants start out much faster at executing the functional-level goals of the move unit-task. However, as with the land unit-task, the relative differences in the latencies among the functional-level goals, i.e., that finding and selecting planes and hold-positions takes longer than moving to those planes and hold-positions, are identical to the Ackerman data and are consistent with our hypothesis from the task analysis.

Functional-level goals of the queue unit-task. Figure 13c plots the mean latencies of the functional-level goals of the queue unit-task and their best fitting power functions. They can be compared to those in Fig. 9c. The overall

χ^2 measure of fit is 55.84 with 64 degrees of freedom, which is not significant. Similar to the functional-level goals of both land and move unit-tasks, the scale factors estimated for our data are much smaller than for those estimated for the Ackerman data, indicating that our participants start out much faster at executing the functional-level goals of the queue unit-task. However, the relative differences in the latencies among the functional-level goals are identical to those in the Ackerman data, with the exception of *Select Hold.* For our participants, the *Select Hold* becomes faster than the *Move to Hold* after Trial 3.

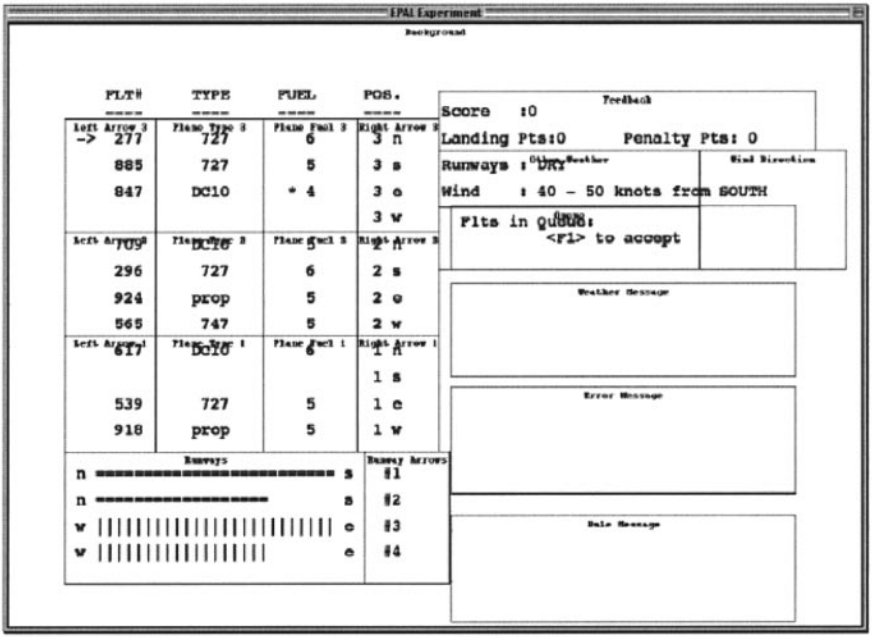
Conclusions. Overall, the mean scale-factor for the functional-level goals in our study was 411 ms compared to 1379 ms in the Ackerman data. In contrast, the mean asymptote in our study was 346 ms compared to 186 ms in the Ackerman data. We think the asymptotes are artificially high in our study because the people in our study were reaching the point where they were waiting for the runways to clear and had nothing else to do. One piece of evidence for this is that there is basically no correlation between the asymptote parameters of the two studies ($r = 0.140$), whereas the scale parameters of the two studies are strongly correlated ($r = 0.927$).

While our participants are faster, we feel that we have generally replicated the learning patterns of the Ackerman data. We have speculated about why our participants are so much faster than Ackerman's participants. Our current hypothesis is that our experiment was run more than a decade later with students who have grown up with computers and computer interfaces, and hence our participants are more sophisticated with computer interfaces. As is shown, much of the learning that is taking place, even for our participants, reflects learning about the particular interface of the KA-ATC task.

Our analysis of the latencies of the unit-tasks and the functional-level keystrokes of the eye-tracking participants clearly mirror our conclusion from the reanalysis of the Ackerman data that we provided previously. Namely our reducibility thesis is supported by the fact that the learning at the different levels of the decomposition is well fit by power functions. Again, this is the *consistent learning* that we identified as the critical test of the Reducibility Hypothesis.

Analysis of the Eye Movement Data

Fig. 15 displays the decomposition of the KA-ATC task screen for the purpose of analyzing where people were gazing. We initially divided the screen into 21 regions of interest. These included 12 regions in the hold, divided into 3 sets of 4 regions for each hold-level. Within a hold-level the regions were divided into the left-arrow column, the plane-type column, the plane-fuel column, and the right-arrow column. The fixations in the left- and the right-arrow columns were due primarily to tracking of the cursor as people pressed the arrow keys to move it up and down. The fixations to the plane-type column were critical because this information was required for



	FLT#	TYPE	FUEL	POS.
->	277	727	6	3 n
	885	727	5	3 s
	847	DC10	* 4	3 e
				3 w
	709	DC10	5	2 n
	296	727	6	2 s
	924	prop	5	2 e
	565	747	5	2 w
	617	DC10	6	1 n
				1 s
	539	727	5	1 e
	918	prop	5	1 w

```
n ========================== s   #1
n ===================        s   #2
w |||||||||||||||||||||||||| e   #3
w ||||||||||||||||||         e   #4
```

Score :0
Landing Pts:0 Penalty Pts: 0
Runways : DRY
Wind : 40 – 50 knots from SOUTH
Flts in Queue:
<F1> to accept

FIG. 15. A decomposition of the Kanfer–Ackerman ATC task into regions for gaze analysis.

knowing which runway the plane could land on (i.e., Rule 4). The fixations to the plane-fuel column were important because this information was required for avoiding plane crashes and penalties for low fuel levels. The runway was divided into 2 regions, one for the runways and the other for the arrows. The right half of the screen was divided into 7 regions corresponding to where different information was presented. In addition to these 21 regions, we separately tracked two additional categories of fixations: (1) fixations to the periphery of the task window, but on the monitor, and (2) fixations off the monitor (e.g., fixations to the keyboard). For on-screen fixations we identified each fixation on the basis of a high-velocity saccade to a region. We attributed both the saccade time and the fixation time to that region. For off-screen fixations, we simply logged the total time spent off the monitor. Thus, we can take the latencies displayed in Fig. 13 and decompose them into the time spent looking at the 23 regions (21 task-regions + 2 additional categories that we described above). However, in some analyses, we collapsed the 23 regions into a smaller set of aggregated regions for tractability. The main reason for collapsing the 23 regions was because many of the regions simply did not have a sufficient number of fixations for our analysis. For each unit-task, we used different aggregations to highlight the regions of relevance and irrelevance specific to that unit-task. We now present the eye-movement analysis for each unit-task.

Land unit-task. For the land unit-task, we aggregated all fixations in hold-levels 2 and 3 into a single category, *Levels 2&3,* to measure how much attention was being spent on the irrelevant hold-levels. Within hold-level 1, we aggregated the left- and right-arrow columns into a single category, *Level 1 Arrow*, to measure how much time people spent monitoring the arrow movements. We separately measured the time spent fixating in the plane-fuel column and the plane-type column in hold-level 1 as *Level 1 Fuel* and *Level 1 Type*. We aggregated fixations on the runways and the runway arrows into one category, *Runways,* and we also aggregated all the remaining on-screen and off-screen fixations into a single category, *Remaining.* Hence, our analysis consisted of six aggregated categories as follows: *Level 1 Type, Level 1 Fuel, Runways, Levels 2&3, Level 1 Arrow,* and *Remaining.*

Of these six categories, we hypothesized that fixations to *Level 1 Type, Level 1 Fuel,* and *Runways* were task-relevant and fixations to *Levels 2&3, Level 1 Arrow,* and *Remaining* were task-irrelevant. *Level 1 Type* is important in the land unit-task because it is required in deciding which runway to land on. *Level 1 Fuel* is important for prioritizing which plane to land, especially early in the experiment when people are slow enough to be assessed low-fuel penalties. Similarly, *Runways* is important because one must monitor the runway to see when it is free. Therefore, we expected to see relatively little decrease in the total fixation time to the *Level 1 Type, Level 1 Fuel,* and *Runways* categories, whereas for the remaining three categories, *Levels 2&3, Level 1 Arrow,* and *Remaining,* we expected to see relatively large decreases in the total fixation time across 18 trials.

Fig. 16a plots the mean of the total time spent fixating in the six aggregated regions and their best fitting power function with the exponent constrained to 1.329. The overall χ^2 measure of fit is 92.879, which is not significant with 96 degrees of freedom. Consistent with our hypothesis, much of the speed-up came from the reduction in the time spent fixating on task-irrelevant regions (i.e., *Level 1 Arrow, Levels 2&3,* and *Remaining*), whereas the time spent fixating on task-relevant regions (i.e., *Level 1 Fuel, Level 1 Type,* and *Runways*) remained relatively unchanged across the trials. In addition, the asymptotes of the power-function fits for the task-relevant regions were higher and their scale factors much lower compared to those for the task-irrelevant regions. It is clear that some regions show greater reductions in fixations than do others.

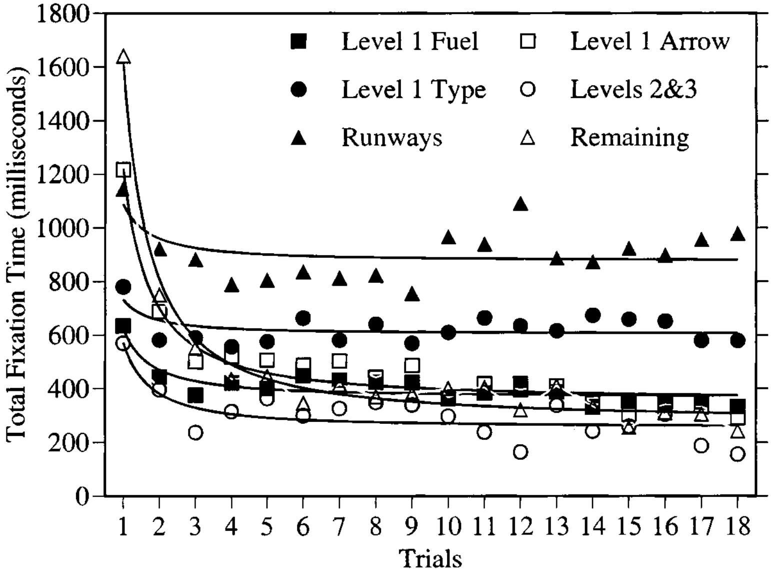
To get a sense for the significance of the different amounts of learning displayed in Fig. 16a, we fitted power functions to each participant to get a separate scale factor for each person, again holding the exponent at 1.329. We then performed *t* tests to determine which scale factors were significantly greater than 0, since this is the component that gets reduced with practice. The scale factors were significantly greater than 0 for *Level 1 Arrows,* $t(9) = 4.727$, $p < .001$, *Levels 2&3,* $t(9) = 5.329$, $p < .001$; *Remaining,* $t(9) = 2.703$, $p < .05$; and *Level 1 Fuel,* $t(9) = 2.461$, $p < .05$. However, it was not significant for *Level 1 Type*, $t(9) = 0.977$, $p > .1$; or *Runways*, $t(9) = 1.196$, $p > .1$. Thus, people appear to be reducing their fixations on all task-irrelevant regions when landing a plane while maintaining the amount of time spent on task-relevant regions, with the exception of the *Level 1 Fuel*. The *t*-test for the *Level 1 Fuel* indicates that the total fixation time to this region is being reduced with practice. While this was not initially expected, we believe this results from the fact that one must monitor fuel levels only if one is slow in landing planes and is in danger of getting low-fuel penalties. But, if one is landing planes rapidly, the fuel level can be more or less ignored. Since our participants are especially fast to begin with, the importance of the fuel level for them is diminished.

Figure 16a captures the change in the amount of fixations in the regions. Figure 16b is an attempt to illustrate the concentration of fixations in particular regions at the end of the experiment for the land unit-task. The numbers in the regions represent the mean amount of time people fixate in those regions per square of visual angle during the last three trials (i.e., Trials 16–18). The grayscales represent visually the amount of time spent in a region relative to the maximum time spent in a region. Hence for Fig. 16b, the grayscales were scaled relative to the 51 ms per square of visual angle spent in *Plane Type 1* region. As can be seen, people spend most of their time in the regions associated with *Hold Level 1* and the *Runways*.

Move unit-task. For the move unit-task we aggregated all the fixations to the left-arrow, plane-type, plane-fuel, and right-arrow columns in the two levels over which the plane was being moved (i.e., hold-levels 2 and 1 for moves from hold-level 2 to 1 and hold-levels 3 and 2 for moves from hold-level 3 to 2) into *Left Arrow*, *Plane Type*, *Plane Fuel*, and *Right Arrow*, respectively. We aggregated all the fixations in the hold-level that was not involved in the movement (i.e., hold-level 3 for moves from hold-level 2 to 1 and hold-level 1 for moves from hold-level 3 to 2) into a single category, *Other Levels*. We aggregated the remaining fixations into a single category, *Remaining*. Hence, our analysis consisted of six aggregated categories: *Plane Type, Plane Fuel, Left Arrow, Right Arrow, Other Levels,* and *Remaining*. For the move unit-task, we expected that the two categories, *Plane Fuel* and *Plane Type,* were relevant, while the other four categories, *Left Arrow, Right Arrow, Other Levels,* and *Remaining,* were not.

Figure 17a plots the mean of the total time spent fixating in the six aggregated regions and their best fitting power function fits with the exponent constrained to 1.329. The overall χ^2 measure of fit is 95.029, which is not significant with 96 degrees of freedom. Consistent with our hypothesis, much of the speed-up came from the reduction in the time spent fixating on task-irrelevant regions (i.e., *Left Arrow, Right Arrow, Other Levels,* and *Remaining*). But for the two task-relevant regions, only the time spent fixating on the *Plane Type* region remains relatively unchanging, while the time spent on the *Plane Fuel* seems to be reduced with practice.

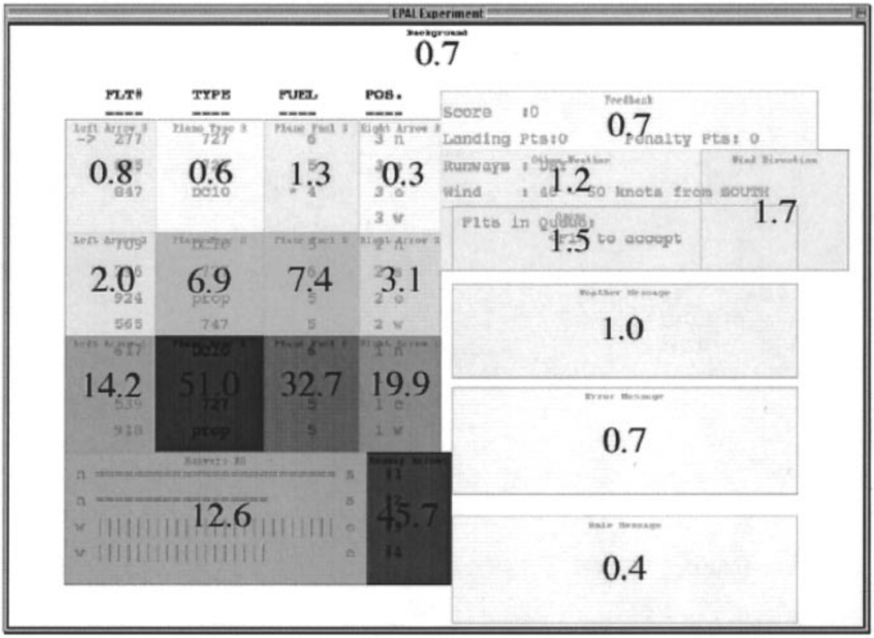
Unit-Tasks	Power Functions	R^2	χ^2	*S.E.*
Level 1 Fuel	$T = 369 + 256\ N^{-1.329}$	0.759	13.920	38
Level 1 Type	$T = 605 + 128\ N^{-1.329}$	0.302	11.598	55
Runways	$T = 876 + 212\ N^{-1.329}$	0.242	18.651	84
Level 1 Arrow	$T = 357 + 856\ N^{-1.329}$	0.942	15.844	52
Levels 2 & 3	$T = 255 + 317\ N^{-1.329}$	0.605	30.046	46
Remaining	$T = 280 + 1329\ N^{-1.329}$	0.978	2.820	116



(a)

FIG. 16. (a) The mean fixation duration per unit-task in task-relevant and task-irrelevant regions during land unit-task and (b) a graphical representation of the distribution of attention over the screen during a land unit-task at the end of the experiment.

To get a sense for the significance of the reductions, we fitted power functions to each participant to get a separate scale parameter for each person, again holding the exponent at 1.329. We then performed *t*-tests to determine which scale factors were significantly greater than 0. As we expected, they were all significant for the task-irrelevant regions: *Left Arrow,* $t(9) = 5.062$, $p < .001$; *Right Arrow,* $t(9) = 4.776$, $p < .001$; *Other Levels,* $t(9) = 2.891$, $p < .01$; and *Remaining,* $t(9) = 2.396$, $p < .05$. And, as expected, the *Plane Type* region was not significant, $t(9) = 0.416$, $p > .1$. Again, contrary to our initial expectation, the *Plane Fuel* region was significant: *Plane Fuel,* $t(9) = 3.726$, $p < .01$, indicating a significant reduction in total fixations in the *Plane Fuel* region. However, as we have explained previously, the monitoring of the fuel levels is important only if one is slow and is in danger



Score :0

Landing Pts:0 Penalty Pts: 0

Runways : Dry

Wind : 40-50 knots from SOUTH

Flts in Queue:

F12 to accept

(b)

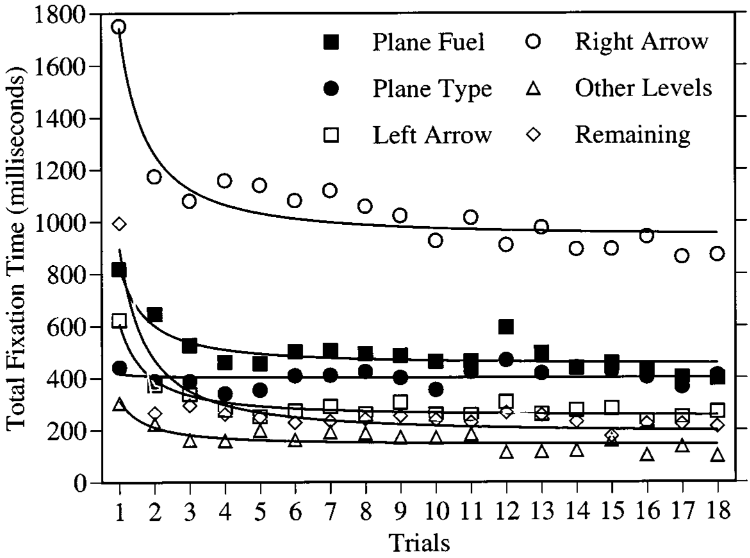
FIG. 16—*Continued*

of crashing planes or getting low fuel-level penalties. Since our participants are fast to begin with, the fuel level is much less important for their performance.

Figure 17b graphically illustrates the distribution of attention across the screen after participants have become skilled. In Fig. 17b, we examine the distribution of attention across the task screen while performing a move unit-task from hold-level 2 to hold-level 1 after they have become skilled in the KA-ATC task. As before, the numbers in the regions represent the mean amount of time people fixate in those regions per square degree of visual angle during the last three trials (i.e., Trials 16–18). The grayscales represent visually the amount of time spent in a region relative to the maximum time spent in a region. Hence for Fig. 17b, the grayscales were scaled relative to the 68 ms per square degree of visual angle spent in right-arrow column in hold-level 1. As can be seen, people spend most of their time in the regions associated with hold-level 2 and hold-level 1, the source and the destination levels for the move unit-task that is being plotted.

Overall, as people become more skilled, they appear to be reducing the time they fixate on all the task-irrelevant regions while maintaining the amount of time spent on the task-relevant regions in the move unit-task. Curiously, while the reduction in the time spent on the *Right Arrow* region is significant, the asymptote is still quite high, at over 800 ms and as Fig.

Unit-Tasks	Power Functions	R^2	χ^2	*S.E.*
Plane Fuel	$T = 449 + 380\,N^{-1.329}$	0.798	14.204	49
Plane Type	$T = 400 + 17\,N^{-1.329}$	0.014	15.395	35
Left Arrow	$T = 249 + 360\,N^{-1.329}$	0.935	12.956	26
Right Arrow	$T = 937 + 808\,N^{-1.329}$	0.875	22.073	63
Other Levels	$T = 142 + 171\,N^{-1.329}$	0.671	20.671	26
Remaining	$T = 183 + 717\,N^{-1.329}$	0.881	9.730	82

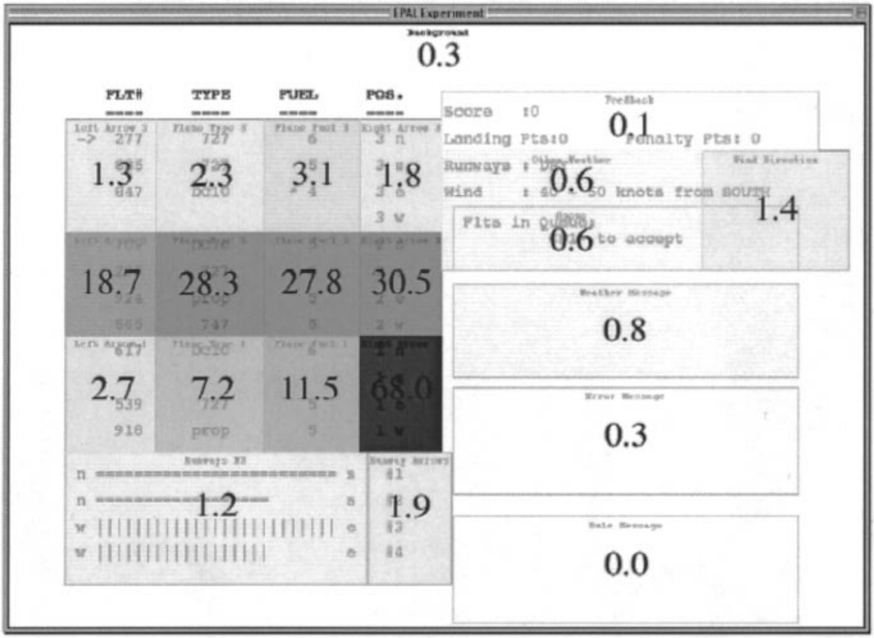


(a)

FIG. 17. (a) The mean fixation duration per unit-task in task-relevant and task-irrelevant regions during move unit-task and (b) a graphical representation of the distribution of attention over the screen during a move unit-task at the end of the experiment.

17b illustrates, it receives the most concentrated visual attention. Clearly, people still feel the need to track the arrow as they move it down to the location of the hold-position for the plane to be moved into, even as they become skilled in the task.

Queue unit-task. For the queue unit-task, we first distinguish between task-relevant and task-irrelevant hold-levels. Specifically, we assume that the target hold-level for the queue unit-task is relevant and the other two remaining hold-levels are irrelevant. We aggregated all the fixations on the left- and the right-arrow columns in the target hold-level into the category *Arrows* and aggregated all fixations on the plane-type and the plane-fuel columns in the target hold-level into the category *Type&Fuel.* All the fixations in nontarget hold levels were aggregated into the category *Other Levels.* In addition, we aggregated fixations to the queue into the category *Queue.* We aggregated



Score :0

Landing Pts:0 Penalty Pts: 0

Wind : 40 - 50 knots from SOUTH

Flts in Queue:

to accept

(b)

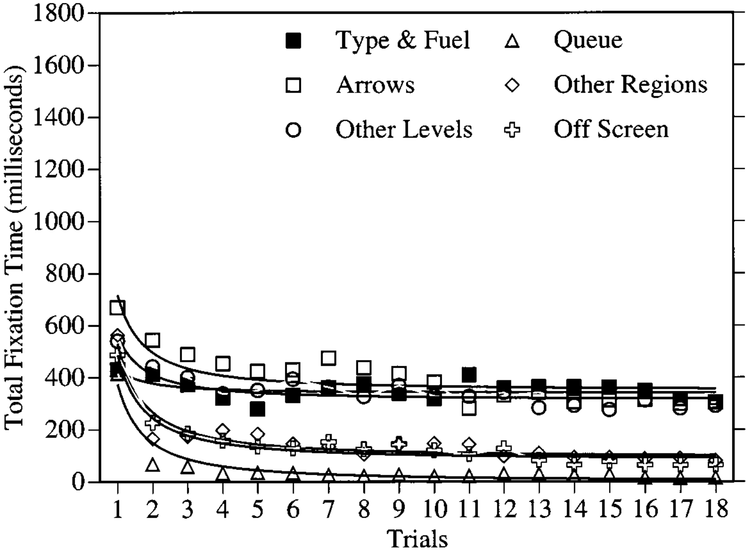
FIG. 17—*Continued*

the remaining on-screen fixations into the category *Other Regions* and the remaining off-screen fixations into the category *Off Screen*. Hence we were left with six categories as follows: *Type&Fuel*, *Arrows*, *Other Levels*, *Queue*, *Other Regions*, and *Off Screen*. Of these six categories, we hypothesized that only the *Type&Fuel* region would be task-relevant.

Figure 18a plots the mean of the total time spent fixating in the six aggregated regions (*Type&Fuel*, *Arrows*, *Other Levels*, *Queue*, *Other Regions*, and *Off Screen*) and their best fitting power functions with the exponent constrained to 1.329. The overall χ^2 measure of fit is 106.321, with 96 degrees of freedom, which is not significant. Consistent with our hypothesis, much of the speed-up comes from the reduction in the time spent fixating on task-irrelevant regions (i.e., *Arrows*, *Other Levels*, *Queue*, *Other Regions*, and *Off Screen*), whereas the time spent fixating on the task-relevant region (i.e., *Type&Fuel*) remains relatively unchanged. The *Queue* region provides a remarkably clear example of a reduction of fixations on a task-irrelevant region. As can be seen, people discover rather quickly that the *Queue* region is irrelevant and basically do not look there after Trial 1.

To get a sense for the significance of the reductions of fixations, we fitted power functions to each participant to get a separate scale parameter for each person, again holding the exponent at 1.329. We then performed *t*-tests to

Unit-Tasks	Power Functions	R^2	χ^2	*S.E.*
Type & Fuel	$T = 340 + 96\ N^{-1.329}$	0.325	15.916	34
Arrows	$T = 349 + 369\ N^{-1.329}$	0.718	33.957	38
Other Levels	$T = 312 + 250\ N^{-1.329}$	0.791	10.807	38
Queue	$T = 0 + 375\ N^{-1.329}$	0.922	23.064	22
Off Screen	$T = 83 + 404\ N^{-1.329}$	0.944	8.643	33
Other Regions	$T = 92 + 439\ N^{-1.329}$	0.904	13.934	37

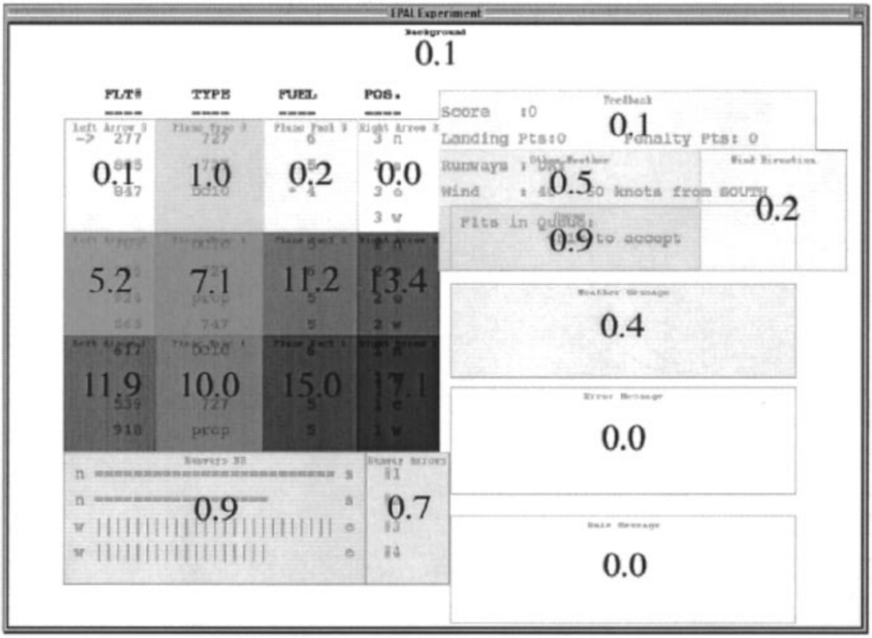


(a)

FIG. 18. (a) The mean fixation duration per unit-task in task-relevant and task-irrelevant regions during queue unit-task and (b) a graphical representation of the distribution of attention over the screen during a queue unit-task at the end of the experiment.

determine which scale factors were significantly greater than 0. As we expected, they were all significant for the task-irrelevant regions: *Arrows*, $t(9) = 3.760$, $p < .01$; *Other Levels*, $t(9) = 2.643$, $p < .05$; *Queue*, $t(9) = 4.826$, $p < .001$; *Other Regions*, $t(9) = 3.954$, $p < .01$; and *Off Screen*, $t(9) = 3.505$, $p < .01$. And, as expected, the task-relevant region, *Type&Fuel*, was not significant, $t(9) = 1.466$, $p > .05$. Again, as people become more skilled, it appears they reduce the amount of time spent on the task-irrelevant regions while maintaining the amount of the time spent on the task-relevant region.

Figure 18b graphically illustrates the distribution of attention over the screen after participants have become skilled. As before, the numbers in the regions represent the mean amount of time people fixate in those regions per square of visual angle during the last three trials (i.e., Trials 16–18). The grayscales represent visually the amount of time spent in a region relative



(b)

FIG. 18—*Continued*

to the maximum time spent in a region. Hence for Fig. 18b, the grayscales were scaled relative to the 17 ms per square of visual angle spent in the right-arrow column in hold-level 1. As can be seen, people spend the majority of their time in hold-level 1, the target hold-level for the queue unit-task. However, they still spend a significant amount of time in hold-level 2. There may be several reasons why people do this. First, the queue unit-task may have been chosen after considering planes in hold-level 2. That is, after considering planes in hold-level 1 and hold-level 2, they ultimately chose to bring a plane from the queue into hold-level 1. Second, as they perform the queue unit-task, they may be planning for future unit-tasks that involve hold-level 2. And third, some fixations that were attributed to hold-level 2 may actually be fixations to the upper hold-position in hold-level 1. This is because hold-level 1 and hold-level 2 are right next to each other. Regardless, the point here is that the majority of fixations during a queue unit-task to hold-level 1 are to the regions associated with hold-level 1.

Summary

Clearly, people attend more and more to the task-relevant regions while learning to ignore the task-irrelevant regions. Figure 19 provides a summary

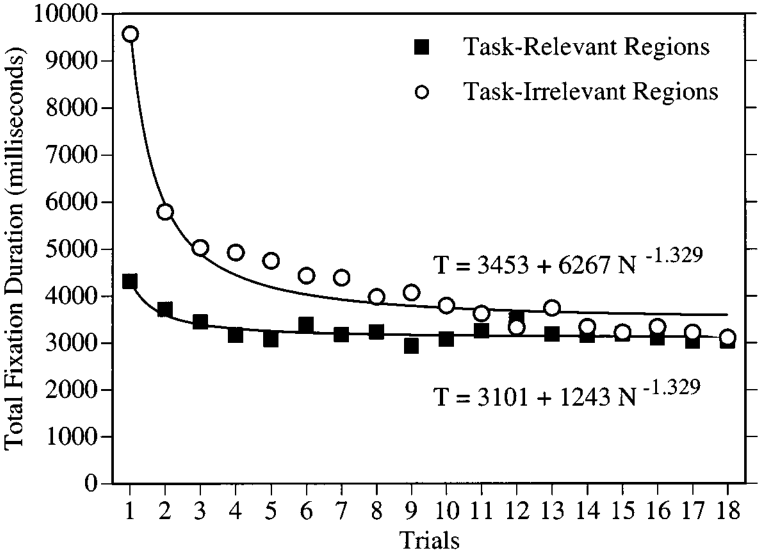


FIG. 19. The mean fixation times in task-relevant and task-irrelevant regions per plane landed in a trial.

of this. In Figure 19, we have classified the fixations to the plane-fuel and plane-type in the hold-level(s) relevant to the three unit-tasks and the fixations on the runways during the land unit-task as *Task-Relevant.* We classified all the remaining fixations as *Task-Irrelevant.* That is, we have aggregated all the fixations classified as task-relevant for the three unit-tasks into a single category, *Task-Relevant,* while aggregating all the fixations classified as task-irrelevant for the three unit-tasks into a single category, *Task-Irrelevant.* We then calculated the amount of time spent in these two categories per plane landed.

As can be seen in Fig. 19, people clearly show a much larger reduction in the time spent fixating on the *Task-Irrelevant* regions relative to the *Task-Relevant* regions. Note that there is still some reduction of time spent in the *Task-Relevant* region. This is expected since there is no guarantee that all fixations in these regions are to task-relevant information. In addition, the fitted power function for the *Task-Irrelevant* category is going down to a nonzero asymptote of 3453 ms. One possible reason for the nonzero asymptote is that some information in the regions that we have aggregated into the *Task-Irrelevant* category is actually task-relevant, such as the need to occasionally monitor the weather information. Also, it is possible that people may be seeking information to plan for the next unit-task while performing the current one. On the other hand, the non-zero asymptote may also be a consequence of the structure of the task. Specifically, since planes take 15 s to taxi down a runway and since there are only two runways available, it is difficult to land planes faster than one every 7.5 s. The time for successful landings on Trial 18 (summing up the *Task-Relevant* and the *Task-Irrelevant* latencies from Figs. 16a–18a) is approximately 6.26 s. The reasons why this time can be below 7.5 s are that (a) there are some unsuccessful unit-tasks that we have not entered into our analysis and (b) people can occasionally have planes on more than two runways when the wind direction changes. Regardless, the important observation here is that people reach the point where it is no longer possible for them to perform faster because of the limitation on performance imposed by the task environment, i.e., when both runways are occupied. During these idle times, they have time on their hands and their eyes may wander to irrelevant portions of the screen.

While idle time may explain why people are looking at irrelevant regions at the end of the experiment, it does not explain why they are spending 10 s looking at these regions at Trial 1. There are at least two possible reasons. First, at the beginning, people may not necessarily know where the relevant information is, and they may be searching the screen for the relevant information. Second, they may be trying to decide what to do next, e.g., considering the possibility of moving a plane between hold-levels or getting a plane from the queue before finally deciding to land a plane. With practice, however, they learn a policy for what to do next without extensively assessing various possibilities.

Other learning trends reflect things more specific than just learning where the information is on the screen and what to do next. We note four as follows:

1. In all three unit-tasks, people showed marked reduction in monitoring the arrow movements. This reflects a switch from monitoring the individual arrow movements to a process of issuing multiple key presses and only monitoring their effect. However, participants still do substantial monitoring of the arrow movements at the end of the experiment.

2. Even though we did not predict it on the basis of task analysis, all three unit-tasks show some reduction of fixations on the fuel level. The fuel level is important early on, e.g., as a method for prioritizing which plane to land next, when people are performing slowly and are in danger of suffering penalties for low fuel or crashing the plane. However, as they become more proficient, a vigilant monitoring of the fuel level of the planes becomes much less critical, since the danger of low fuel penalties are minimal.

3. Initially, when people are learning to bring a plane from the queue, they fixate on the queue to check that there is a plane available. However, they quickly learn that there are always planes in the queue and that checking the queue is unnecessary (Fig. 18a).

4. Figure 18a also illustrates a dramatic reduction in off-screen fixations for entering planes from the queues. We assume that people are learning the location of the F1 key with practice and do not have to visually guide the pressing of this key.

CONCLUSIONS

The analyses we have reported generally support the view that underlying the improvement in the performance of the Kanfer–Ackerman Air Traffic Controller Task is a power-law speed-up of performance of the unit-tasks and that the improvement on these unit-tasks can be decomposed into improvements on simple actions. In this article we showed that, for both the original Ackerman data and our data, people could conceivably be eliminating all cognitive time and simply spending the time necessary to execute the keystrokes. In our experiment, we showed that much of this cognitive time consisted of fixation time and that the reduction in the cognitive time involved the reduction in fixations to task-irrelevant information on the screen. Participants apparently move to a point where their speed at doing the task is largely determined by the time needed to encode the information, although they appeared to have time to spare to engage in some irrelevant fixations.

These two keystroke and eye-movement characterizations might at first seem to be at odds. What is the nature of the residual time on the task—to execute keystrokes or attend to needed information? These two characterizations can be reconciled by realizing that the task poses perceptual, motor, and cognitive demands on people. It is possible to engage in cognitive, perceptual, and motor activities in parallel but have each of these streams be serial within itself. That is, at any point in time we can visually attend to only one thing, such as performing a single action or engaging in a single line of thought. The notion that these three streams are serial within themselves but run in parallel was introduced by the CPM GOMS model (Gray, John, & Atwood, 1993) and has been adopted by ACT-R/PM (Byrne & Anderson, 1998). It is also found in EPIC (Meyer & Kieras, 1997), except that EPIC does not have the constraint of serial cognitive processing. At any point in time, one of these three processes, perceptual-attentional, motor, or cognitive, can be on the critical path with other processes waiting for its outcome. It is also possible for the task environment to come to be on the critical path, as when a person is waiting for a runway to clear. There is relatively little learning of the motor component but we postulate considerable learning of the perceptual and the cognitive components. With practice, the perceptual encoding time reduces to a nonzero minimum even as the motor execution has a nonzero minimum. We speculate that with enough practice non perceptual cognitive activities would almost totally disappear from the critical path.

Our eye-movement analysis raised the question concerning the degree to which the cognitive component is on the critical path at the beginning of the task. The speed-up in this task can be conceived of as entirely eliminating unnecessary encoding time. The fact that 85% of the speed-up involves reduction of fixations in irrelevant regions is consistent with this view. While we think the changes in the fixation pattern causes the speed-up, it is worth considering an alternative possibility, which is that the speed-up causes the changes in the fixation pattern. According to this argument, the cognitive system is typically on the critical path, and while people are engaged in thinking about the task, their eyes are free to wander about the screen and elsewhere. As the irrelevant regions of the screen are the larger fraction of the screen, a larger portion of the speed-up would be attributed to those regions. This does not indicate that people are learning not to look at these regions.

While there may be some truth in this alternative explanation, it cannot be the entire story, nor do we think that it is even the majority of the story. First, it is incapable of explaining some of the specific changes in fixations, such as the large reduction in fixations to the queue region. Moreover, if it were the majority of the explanation, it would predict that comparably sized regions would show comparable speed-up, since they would be just as likely to be fixated in the eye's random wanderings. However, as we have demonstrated, there were almost no reduction in fixations to the plane-type regions in the relevant levels, while there were larger reductions in the fuel area and even larger reductions in the regions of the arrows despite the fact that these regions were of comparable physical size. Thus, people disproportionately reduce their fixations to regions that are less task-relevant.

If the majority of the improvement in this task is due to more efficient attention, then there remains the issue of the underlying mechanism that drives this attentional learning process. When asked what they learned, people reported their general strategy for doing the task, such as entering planes only into hold-level 1. Additionally, they reported some things that seem related to their improvement, such as not having to worry about planes in the queue and not having to worry about the fuel level of the planes. On the other hand, they did not report that they monitored the arrow movement less or that they learned where the F1 key was, although they might have admitted to this if queried. They might even not be aware of the fact that they spent less time looking at irrelevant regions of the screen. Thus, much of the learning that leads to expertise at this task is not something about which people are particularly aware. It occurs at a lower level than the one that people think about when trying to understand or explain their performance in the task. In this sense, we think the task is representative of many skills in which people gradually acquire fluency. That is, the learning involves attentional shifts of which the participants are not even aware.

To return to the different types of skill-acquisition theories described in the introduction, we suspect that different parts of the improvement in the task are due to different mechanisms. People change their strategy for performing the task, such as by not checking the queue or the weather conditions. This would be an instance of a procedure selection model. They also strengthen their memories for the location of critical information and task procedures. This would be an instance of a strengthening model. Additionally, they also appear to form macro operators, such as by hitting four down arrow keys, which means that individual keystrokes no longer have to be selected. This would be an instance of a procedure transformation model. Thus, this research does not select among the types of theories of skill acquisition, and we are inclined to believe that multiple learning mechanisms are at work. Rather, this research indicates that skill acquisition mechanisms work; they achieve their effect by reducing the time needed to perform the individual steps of the task.

At the beginning of this article we defined ‘‘consistent learning’’ as the criterion for accepting the reducibility hypothesis. We think our analysis of the data in this task certainly satisfies the criterion of consistent learning. Not only did we show that all the individual keystrokes speed up without discontinuities, we also showed that to a very good approximation they speed up with a common exponent and common asymptote. Both of these parameter constraints tells us something important. The exponent basically captures the shape of the learning curve. Thus, to the extent that these curves shared the same exponent we have evidence that learning is happening at the same rate for each step. We should point out that a theory like ACT-R (Anderson & Lebiere, 1998), which attributes such speed-ups to basic strengthening processes in the architecture, would predict such a common exponent. Second, to a good approximation the learning curves appear to share a common asymptote and this asymptote takes on the value of the minimum keying time. This suggests that the learning is taking subjects to a point where the motor component is the rate-limiting aspect of their performance. All of the cognitive and attentional time, represented by the scale factors, is eliminated.

REFERENCES

Ackerman, P. L. (1988). Determinants of individual differences during skill acquisition: Cognitive abilities and information processing. *Journal of Experimental Psychology: General,* **117,** 288–318.

Ackerman, P. L., & Kanfer, R. (1994). *Kanfer-Ackerman air traffic controller task© CD-ROM database, data collection program, and playback program:* Office of Naval Research, Cognitive Science Program.

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

Anderson, J. R., & Lebiere, C. (1998). *The Atomic Components of Thought.* Mahwah, NJ: Erlbaum, Inc.

Anderson, J. R. (1989). Practice, working memory, and the ACT* theory of skill acquisition: A comment on Carlson, Sullivan, and Schneider (1989). *Journal of Experimental Psychology: Learning, Memory, and Cognition,* **15,** 527–530.

Anderson, J. R., Conrad, F. G., & Corbett, A. T. (1989). Skill acquisition and the LISP tutor. *Cognitive Science,* **13,** 467–505.

Anderson, J. R., Fincham, J. M., & Douglass, S. (1999). Practice and retention: A unifying analysis. *Journal of Experimental Psychology: Learning, Memory, and Cognition,* **25,** 1120–1136.

Anderson, R. B., & Tweney, R. D. (1997). Artifactual power curves in forgetting. *Memory & Cognition,* **25,** 724–730.

Ashcraft, M. H. (1992). Cognitive arithmetic: A review of data and theory. *Cognition,* **44,** 75–106.

Byrne, M. D., & Anderson, J. R. (1998). Perception and Action. In J. R. Anderson & C. Lebiere (Eds.), *The atomic components of thought* (pp. 167–200). Mahwah, NJ: Erlbaum, Inc.

Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human-computer interaction.* Hillsdale, NJ: Erlbaum.

Crossman, E. R. F. W. (1959). A theory of the acquisition of speed-skill. *Ergonomics,* **2,** 153–166.

Delaney, P. F., Reder, L. M., Staszewski, J. J., & Ritter, F. E. (1998). The strategy-specific nature of improvement: The power law applies by strategy within task. *Psychological Science,* **9,** 1–7.

Derks, P. L. (1974). The length–difficulty relation in immediate serial recall. *Journal of Verbal Learning & Verbal Behavior,* **13,** 335–354.

Douglass, S. (1998). *EPAL: Data collection and analysis software for eye-tracking experiments.* Pittsburgh, PA: Carnegie Mellon University.

Epelboim, J., Kowler, E., Edwards, M., Collewijn, H., Erkelens, C. J., & Steinman, R. M. (1994). *Natural oculomotor performance in looking and tapping tasks.* Paper presented at the Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society.

Gray, W. D., John, B. E., & Atwood, M. E. (1993). Project Ernestine: Validating a GOMS analysis for predicting and explaining real-world task performance. *Human Computer Interaction,* **8,** 237–309.

Haider, H., & Frensch, P. A. (1996). The role of information reduction in skill acquisition. *Cognitive Psychology,* **30,** 304–337.

Haider, H., & Frensch, P. A. (1999). Eye movement during skill acquisition: More evidence for the information-reduction hypothesis. *Journal of Experimental Psychology: Learning, Memory, & Cognition,* **25,** 172–190.

Heathcote, A., Brown, S., & Mewhort, D. J. K. (2000). *The power law repealed: The case for an exponential law of practice.* Psychonomic Bulletin & Review, **7,** 185–207.

Kowler, E. (1995). Eye movements. In S. M. Kosslyn & D. N. Osherson (Eds.), *Visual cognition: An invitation to cognitive science* (pp. 215–265). Cambridge, MA: The MIT Press.

Lee, F. J., Anderson, J. R., & Matessa, M. P. (1995). *Components of dynamic skill acquisition.* Paper presented at the Proceedings of the Seventeenth Annual Conference of the Cognitive Science Society.

Lemaire, P., & Siegler, R. S. (1995). Four aspects of strategic change: Contributions to children's learning of multiplication. *Journal of Experimental Psychology: General,* **124,** 83–97.

Lewis, C. H. (1978). *Production system models of practice effects.* Ph.D. thesis, University of Michigan, Ann Arbor.

Logan, G. D. (1988). Toward an instance theory of automatization. *Psychological Review,* **95,** 492–527.

MacKay, D. G. (1982). The problem flexibility, fluency, and speed-accuracy trade-off in skilled behavior. *Psychological Review,* **89,** 483–506.

Meyer, D. E., & Kieras, D. E. (1997). A computational theory of executive cognitive processes and multiple-task performance. I. Basic mechanisms. *Psychological Review,* **104,** 3–65.

Myles-Worsley, M., Johnston, W. A., & Simons, M. A. (1988). The influence of expertise on x-ray image processing. *Journal of Experimental Psychology: Learning, Memory, and Cognition,* **14,** 553–557.

Myung, I. J., Kim, C., & Pitt, M. (2000). Toward an explanation of the power-law artifact: Insights from response surface analysis. *Memory & Cognition,* **28,** 832–840.

Neves, D. M., & Anderson, J. R. (1981). Knowledge compilation: Mechanisms for the automatization of cognitive skills. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition* (pp. 57–84). Hillsdale, NJ: Erlbaum.

Newell, A., & Rosenbloom, P. S. (1981). Mechanisms of skill acquisition and the law of practice. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition* (pp. 1–55). Hillsdale, NJ: Erlbaum.

Palmeri, T. J. (1999). Theories of automaticity and the power law of practice. *Journal of Experimental Psychology: Learning, Memory, and Cognition,* **25,** 543–551.

Reder, L. M., & Ritter, F. E. (1992). What determines initial feeling of knowing? Familiarity with question terms, not with the answer. *Journal of Experimental Psychology: Learning, Memory, and Cognition,* **18,** 435–451.

Reder, L. M., & Schunn, C. D. (1999). Bringing together the psychometric and strategy worlds: Predicting adaptivity in a dynamic task. In D. Gopher & A. Koriat (Eds.), *Cognitive regulation of performance: Interaction of theory and application: Attention and Performance XVII* (pp. 315–342). Cambridge, MA: The MIT Press.

Rickard, T. C. (1997). Bending the power law: A CMPL theory of strategy shifts and the automatization of cognitive skills. *Journal of Experimental Psychology: General,* **126,** 288–311.

Rickard, T. C., & Bourne, L. E. J. (1996). Some tests of an identical elements model of basic arithmetic skills. *Journal of Experimental Psychology: Learning, Memory, and Cognition,* **22,** 1281–1295.

Rubin, D. C., & Wenze., A. E. (1996). One hundred years of forgetting: A quantitative description of retention. *Psychological Review,* **103,** 734–760.

Shapiro, K. L., & Raymond, J. E. (1989). Training of efficient oculomotor strategies enhances skill acquisition. *Acta Psychologica,* **71,** 217–242.

Siegler, R. S. (1988). Strategy choice procedures and the development of multiplication skill. *Journal of Experimental Psychology: General,* **117,** 258–275.

Vickers, J. N. (1992). Gaze control in putting. *Perception,* **21,** 117–132.

Wixted, J. T., & Ebbesen, E. B. (1991). On the form of forgetting. *Psychological Science,* **3,** 409–415.

Wixted, J. T., & Ebbesen, E. B. (1997). Genuine power curves in forgetting: A quantitative analysis of individual subject forgetting functions. *Memory & Cognition,* **25,** 731–739.

Accepted September 1, 2000
Published online March 20, 2001