The Cambridge Handbook of Computational Psychology

Edited by
RON SUN
Rensselaer Polytechnic Institute

2008

CAMBRIDGE
UNIVERSITY PRESS

CHAPTER 13

Computational Models of Skill Acquisition

Stellan Ohlsson

1. Introduction: Topic, Scope and Viewpoint

Daily life is a sequence of tasks: cook breakfast; drive to work; make phone calls; use a word processor or a spread sheet; take an order from a customer, operate a steel lathe or diagnose a patient; plan a charity event; play tennis; shop for groceries; cook dinner; load the dishwasher; tutor children in arithmetic; make a cup of tea; brush teeth; and set the alarm for next morning. The number of distinct tasks a person learns to perform in a lifetime is certainly in the hundreds, probably in the thousands.

There is no entirely satisfactory way to refer to the type of knowledge that supports task performance. The phrase *know-how* has entered the popular lexicon but is stylistically unbearable. The philosopher Gilbert Ryle (1949/1968) famously distinguished *knowing how* from *knowing that*. Psychometricians talk about *abilities* (Carroll, 1993), whereas artificial intelligence researchers talk about *procedural knowledge* (Winograd, 1975); both terms are somewhat misleading or awkward. The alternative term *practical knowledge* resonates with other relevant usages, such as the verb *to practice*, the cognitive anthropologist's concept of *a practice*, the philosopher's concept of *practical inference*, and the common sense distinction *theory versus practice*. In this review, the term "practical knowledge" refers to whatever a person knows about how to perform tasks, achieve desired effects, or reach goals, whereas "declarative knowledge" refers to knowledge about how things are.

How is practical knowledge acquired? How can a person – or some other intelligent agent, if any – bootstrap himself or herself from inability to mastery? The purpose of this chapter is to organize the stock of current answers to this question in a way that facilitates overview, comparison, and future use.

This chapter focuses on cognitive as opposed to sensori-motor skills. The distinguishing feature of a cognitive skill is that the physical characteristics of the relevant actions (amplitude, force, speed, torque, etc.) are not essential for task performance. Compare tennis with chess in this respect. The success of a tennis serve is a function of the exact trajectory of the racket, but a chess move is the same move, from the point of view of chess, whether it is executed by moving the piece by hand, foot, or mouth, physically very different movements. The equivalence class of movements that count as making *chess move so-and-so* abstracts over the physical characteristics of those movements, and its success, as a chess move, is not a function of those characteristics. Many skills have both cognitive and sensori-motor components, but the hypotheses discussed in this chapter were not designed to explain the acquisition of the latter.

A second boundary of this review derives from its focus on computer models. It is possible and useful to reason with informal hypotheses, but for inclusion here, a hypothesis has to be implemented as a running computer program, and there must be at least one publication that reports results of a simulation run. Also, this chapter emphasizes models that have been proposed as explanations for human learning over contributions to machine learning research or robotics. This chapter focuses on models that create or alter symbolic knowledge representations and deals only briefly with models that learn by adjusting quantitative properties of knowledge structures. Although occasionally refering to empirical studies, this chapter is primarily a review of theoretical concepts. This chapter does not attempt to pass judgment on the empirical adequacy of the different models, for reasons that are spelled out in the last section. Although no current hypothesis explains all skill acquisition phenomena, this chapter proceeds on the assumption that each hypothesis contains some grain of truth to be extracted and incorporated into future models.

The unit of analysis throughout is the individual *learning mechanism*. A learning mechanism is specified by its *triggering conditions*, that is, the conditions under which it will execute, and by the particular *change* that occurs under those conditions. As an illustration, consider the classical concept of association: If two concepts are active simultaneously, a memory link is created between them. The triggering condition is in this case the simultaneous occurrence of the two concepts in working memory; the change is the creation of a link. The learning mechanisms considered in this chapter are considerably more complicated, but their descriptions can nevertheless be parsed into a set of triggering conditions and a change process.

The learning mechanism is a more fine-grained unit than the model or the cognitive architecture because a model might include multiple learning mechanisms and, in fact, some models do. Slicing models into their component learning mechanisms facilitates comparisons among the latter. This chapter does not review every application of every model, but focuses on publications that introduce, explain, or demonstrate learning mechanisms.

Improvements in a skill cannot come out of thin air, so a learning mechanism must draw on some source of information. Different mechanisms operate on different sources: Learning from instruction is not the same process as learning from error. In general, each learning mechanism takes a specific type of information as input. I refer to this as the *Information Specificity Principle*. (see Figure 13.1).

It is highly unlikely that all phenomena associated with the acquisition of cognitive skills can be explained by a single learning mechanism. We do not know how many distinct modes of cognitive change there are, but it is assumed that the observable changes in overt behavior are a product of multiple, interacting mechanisms.

In short, to explain skill acquisition is to specify a repertoire of learning mechanisms, each mechanism consisting of a triggering condition and a change process, to implement these within some performance system and to demonstrate, by running the resulting simulation model, that the cumulative outcome of the interactions among the specified mechanisms mimics the acquisition of ecologically relevant cognitive skills across tasks, initial knowledge states, and learning scenarios. This formulation of the skill acquisition problem is the product of a century of scientific progress.

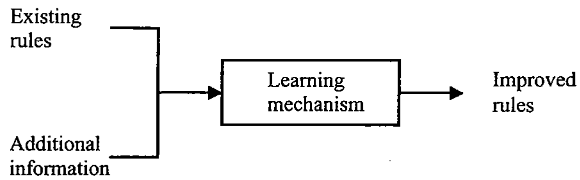

Figure 13.1. Schema for learning mechanisms.

2. History

In William James' (1980) comprehensive summary of the principles of psychology, there is a chapter on habit formation but no chapter on learning. Systematic empirical research on the acquisition of cognitive (as opposed to sensori-motor) skills began with Edward Thorndike's PhD thesis, begun in 1896 under James at Harvard University but issued a few years later from Teachers College at Columbia University. Thorndike (1898) investigated how various species of animals learned to escape from cages with nonobvious door-opening mechanisms. He displayed the time it took individual animals to escape from a box as a function of trial number. Although Hermann Ebbinghaus (1885/1964) had already published curves for the memorization and forgetting of lists of syllables, Thorndike was the first person to plot what we now call practice curves for complex skills. He formulated the Law of Effect, which says that the probability that a learner will perform a particular action is increased when the action is followed by a positive outcome (a "satisfier" in Thorndike's terminology) and decreased when followed by a negative outcome ("annoyer"; Thorndike, 1927). This proved to be an enduring insight.

Learning became the major theme of the behaviorist movement, conventionally dated as beginning with Watson's (1913) article, "Psychology as the Behaviorist Views It." During the 1913–1955 period, *experimental psychology* and *learning theory* became almost synonymous in the United States, but the dominant experimental paradigm for the study of learning was the memorization of lists of letters, syllables, or words. Woodworth's (1938) attempt to replicate James's comprehensive summary from fifty years earlier included a chapter on practice and skill that mentioned twenty-seven studies that tracked learning in complex tasks, like archery, telegraphy, and typing (pp. 156–175). The negatively accelerated shape of the practice curve was well established, and the search for a mathematical equation had begun (pp. 170–173). The idea that the process of acquiring a new skill goes through phases that involve different types of changes was stated but not developed. Both ideas have turned out to be enduring (Ackerman, 1990; Newell & Rosenbloom, 1981).

During World War II, psychologists in Britain and the United States were prompted to move away from list learning and focus on complex skills by the need to contribute to the war effort (Gardner, 1985). The war posed novel problems, such as how to train anti-aircraft gunners. (Anti-aircraft guns were still aimed manually.) A second transforming influence was that psychologists worked alongside engineers, scientists, and mathematicians who were in the process of creating new information technologies. Code breaking and other information processing problems led researchers to realize that information can be measured and processed in objective and systematic ways, making it possible both to build information processing systems and to view humans and animals as examples of such systems.

Immediately after the war, Norbert Weiner at the Massachusetts Institute of Technology envisioned an interdisciplinary science – called *cybernetics* – which was to study complex information-based systems, encompassing humans, machines, and animals, in terms of *feedback circles*. The idea of replacing the stimulus–response reflex with the feedback circle as the central concept of psychology played the star role in Miller, Galanter, and Pribram's (1960) sketch of what we now call the cognitive architecture. Although the concept of feedback remains important, a variety of factors, including Wiener's focus on continuous feedback, which can only be manipulated with complex mathematics, reduced the influence of the cybernetic approach (Conway & Siegelman, 2005). It was soon overtaken by the digital approach, variously called *complex information processing* and, eventually, *artificial intelligence*, launched by Newell, Shaw, and Simon (1958) with an article describing the Logic Theorist, the first symbol-processing computer program that performed a task, logical deduction, that is recognized as requiring intelligence when done by people. The program formalized the notion of *heuristic search*, another enduring concept. Significantly, the article was published in *Psychological Review* rather than an engineering journal, and the authors offered speculations on the relation between their program and human reasoning. The article thus simultaneously established the two fields of artificial intelligence and cognitive modeling (Crevier, 1993).

Paradoxically, the success of the digital symbol manipulating approach had a detrimental effect on the study of learning. In the period 1958–1979, few leading cognitive psychologists studied the effects of practice or other problems related to skill acquisition (but see Welford, 1968, for an exception). The new modeling techniques were at first applied to steady-state performance. This was difficult enough with the crude programming tools available at the time. Success in simulating human behavior – any behavior – was recognized as an achievement in and of itself, even if a model did not simulate changes in that behavior over time.

The era of computational skill acquisition models was inaugurated with a *Psychological Review* article by Anzai and Simon (1979). They presented a computer program that modeled the successive strategy changes of a single person who solved the Tower of Hanoi problem multiple times. The article demonstrated the feasibility of simulating the acquisition and not only the execution of cognitive skills. The article was closely followed by the first set of learning assumptions associated with J. R. Anderson's ACT model. Anderson, Kline, and Beasley (1978) laid out a design for a cognitive architecture with multiple learning mechanisms, later published in Anderson (1982, 1983, 1987, 1993).

Several of the early simulation efforts were formulated within the production system framework (Davis & King, 1977; Neches, Langley, & Klahr, 1987; Newell, 1972, 1973; Newell & Simon, 1972; Waterman & Hayes-Roth, 1978). In this framework, practical knowledge is encoded in *rules*, knowledge structures of the form *if the current goal is* G, *and the current situation is* S, *then consider performing action A*. A production system architecture executes a collection of such rules through a cyclic process: Match the G and S components against the current goal and the current situation (as represented in working memory); enter all matching rules into a *conflict set*; select a rule by resolving the conflict; and execute (the action of) the selected rule. The action alters the state of the world, and the cycle repeats. The production rule notation

Goal, Situation → Action

is as close as the field has come to a *lingua franca* for the analysis of cognitive skills.

The Anzai and Simon (1979) article, the emergence of production systems as a shared formalism, the launching of Anderson's ACT project, and other events collectively triggered an unprecedented explosion of the theoretical imagination. More new hypotheses about the mechanisms behind the acquisition of cognitive skills were proposed in the years 1979–1995 than in the previous century. The success of the initial models established computer simulation as a workable and even indispensable theoretical tool. Informal arguments to the effect that this or that learning mechanism has such-and-such behavioral consequences remain acceptable, but they are clearly inferior to predictions produced by running a simulation model. The last twenty years have seen a proliferation of formal approaches, including neural networks (Chapter 2 in this volume), genetic algorithms (De Jong, 1990; Holland, 1975), and dynamic systems (Chapter 4 in this volume). However, the invention of novel learning mechanisms appears to have slowed.

The following four sections review the skill acquisition mechanisms that have been proposed since Thorndike's experimental subjects clawed, pecked, and pushed their way out of his problem boxes. The explanatory power of these mechanisms disproves the pessimists who would argue that cognitive modeling of learning has made little progress.The task of disproving the optimists is postponed until the last section.

3. How Does Skill Practice Begin?

The three phases of skill acquisition sketched by Woodworth (1938) and articulated further by Fitts (1964) and others provide a useful framework for thinking about skill acquisition. At the outset of practice, the learner's main problem is how to get started, how to construct an initial strategy for the target task. Once the learner is acting vis-à-vis the task, the challenge is to improve that initial strategy until the task has been mastered. Finally, in the long run, the challenge is to optimize the mastered strategy. Each phase provides different sources of information and hence affords different learning mechanisms. This section reviews learning mechanisms that primarily operate within the first phase, while the following two sections focus on the second and third phases. Within each phase, learning mechanisms are distinguished on the basis of the source of information that they draw on, their triggering conditions, and the type of change they compute.

The grouping of learning mechanisms by phase should not be interpreted as a claim that the phases are created by a big switch in the mind that turns mechanisms on and off. I assume that all learning mechanisms operate continuously and in parallel, but the types of information they require as input might vary in abundance and accessibility over time. The phases emerge out of the fact that some types of information becomes less accessible, frequent, or useful as learning progresses, whereas other types of information increase, producing a gradual shift in the relative importance of different types of changes across the successive phases. The final behavior – the fast, accurate, smooth, and nearly effortless expert performance – is the composite and aggregate outcome of the mechanisms operating in all three phases.

For present purposes, the first phase is defined as starting when the learner encounters the task and as ending when the learner completes the task for the first time. The learning mechanisms that dominate this phase are answers to the question, *how can skill practice begin?* How does a learner know what to do before he or she has learned what to do? There are at least four principled approaches to this paradox, corresponding to four distinct sources of information that can be available at the outset of practice: instructions, abstract declarative knowledge, prior skills, and someone else's solution.

3.1. *Interpret Exhortations*

Unfamiliar tasks often come with written or spoken recipes for what to do, variously referred to as *advice* or *instructions*; in linguistic terminology, *exhortations*. Dispensing spoken advice is a large part of what coaches and tutors do. Written sources include cook books, manuals for electronic devices, instruction sheets for assembly-required furniture, and software manuals. Exhortations are presumably understood via the standard discourse comprehension processes studied in psycholinguistics (word recognition, mental lexicon look-up, disambiguation, syntactic parsing, implicit inferences and so on; see Gernsbacher, 1994), but people cannot follow complex instructions without hesitation, backtracking, errors, and repeated rehearsals, even when those instructions are fully understood, so additional processes are required to translate the output of discourse comprehension into executable practical knowledge.

In McCarthy's (1959, 1963)[1] early design for an advice taker system, reasoning about exhortations and actions was assimilated to logical deduction via axioms that define nonlogical operators like *can* and *do*. Instructions are propositional grist for the deductive mill; no special process needed (see also Simon, 1972). This deductive reasoning approach continues within logic programming (Amir & Maynard-Zhang, 2004; Giunchiglia et al., 2004) but remains largely unexplored by psychologists modeling human skill acquisition (but see Chapter 5 in this volume).

The Advice Taker model described by Mostow (1983) and Hayes-Roth, Klahr, and Mostow (1981) was designed to operationalize exhortations by transforming them into executable plans. In the context of the game of hearts, a novice might be told *if you can't take all the points in a round, take as few as possible*. If the learner does not yet know how to take few points, he or she has to refer to the definitions of *take*, *few*, and *points* to expand the advice into an action he or she knows how to do, for example, *play a low card*. This amounts to a top-down search through all alternative transformations allowed by concept definitions, background knowledge, and so on. Mostow (1983) reports using a repertoire of approximately 200 transformation rules to find a 100-step expansion of the advice *avoid taking points* into the executable action *play a low card* (given a particular state of knowledge about the game).

Nonlogical operators and transformation rules have to be general across domains to serve their purpose, so they share the difficult question of their origin. A contrasting approach is employed in Instructo-Soar (Huffman & Laird, 1995). An exhortation is operationalized by constructing an explanation for why it is good advice. The system conducts an internal search (look-ahead) from the current situation (or a hypothetical situation specified in the conditional part of an exhortation like, *if the red light is flashing, sound the alarm*) until it finds a path to the relevant goal that includes the recommended step. Soar's chunking mechanism – a form of explanation-based learning[2] – is then applied to create a new rule (or rules) that can generate that path in the future without search. This technique allows Instructo-Soar to acquire complex actions as well as other types of knowledge from task instructions. Instructo-Soar is equipped with a natural language front end and receives instructions in English. An alternative approach to translating instructions for a radar operting task into production rules in the ACT-R system is described by Taatgen (2005). A simpler translation of instructions into production rules was implemented in the Instructable Production System (Rychener, 1983; Rychener & Newell, 1978).

Doane et al. (2000) described a system, UNICOM, that learns to use the Unix operating system from instructions. An updated version, called ADAPT-PILOT, accurately models the effect of on-line instructions on the behavior of jet pilots during training (Doane & Sohn, 2000; Sohn & Doane, 2002). These models are based on the construction-integration theory of discourse comprehension proposed by Kintsch (1998). General background knowledge and knowledge of the current state of the world are represented as propositions, and *plan elements* – internal representations of executable actions – are represented in

1 The two papers referenced here were reprinted as sections 7.1 and 7.2, respectively, of a chapter titled "Programs with Common Sense" in Minsky (1968). Note that the chapter with that same title in Lifschitz (1990) corresponds to section 7.1, i.e., to McCarthy (1959), but leaves out the content in McCarthy (1963).

2 Explanation-based learning, henceforth EBL, is a machine learning technique that compresses a deductive proof or a sequence of rule executions into a single knowledge structure that connects the premises and the conclusion. The key aspect of the technique is that it aligns variable bindings in the successive steps in such a way as to identify which constants can be replaced by variables. That is, it produces a motivated, conservative generalization of the compressed structure. What kind of learning EBL implements depends on context, origin of its input, and the use made of its output. See Russell and Norvig (1995) for an introduction.

terms of their preconditions and outcomes. All of these are linked in a single associative network on the basis of overlap of predicates. Links can be excitatory or inhibitory. In each cycle of operation, a standard network algorithm is used to compute the current activation level of each node (proposition or plan element). The plan element with the highest activation level is chosen for execution. Its outcome is recorded in the network, and the cycle starts over. Learning occurs by incorporating verbal prompts, for example, *you will need to use the arrow symbol "≥" that redirects the output from a command to a file*, into the associative network. This alters the set of connections, hence, the outcome of the construction-integration process, and, ultimately, which plan element is executed.

There are other applications of the network concept to the problem of learning from instruction. The CAP2 network model described by Schneider and Oliver (1991) and Schneider and Chein (2003) is instructable in the related sense that a symbolic representation of the target skill can inform and speed up learning in a neural network.

The proposed mechanisms capture the complexity of learning from exhortations, but the psychological validity of their details is open to question. Also, these mechanisms apply primarily to initial instructions. They do not model learning from tutorial feedback, because they do not relate what is said to what was just done. Models of learning from instruction are potentially useful in educational research (Ohlsson, 1992; Ohlsson, Ernst & Rees, 1992; VanLehn, Ohlsson, & Nason 1994).

3.2. *Reason from Abstract Declarative Knowledge*

An intelligent agent who desires to travel southward but who is facing north, and who knows something about the compass, should be able to infer that his or her next action ought to be *turn around*. The abstract declarative principles that hold in this situation – the agent's mental model of the Earth, the compass, and their relation – can guide action, and it is tempting to believe that the flexibility of human beings is, in part, a function of cognitive processes that make such guidance explicit. How else did Christoffer Columbus decide that *the Earth is round* implies *sail West*? The strength of the intuition belies the difficulty of specifying the relevant processes.

The *proceduralization* mechanism proposed by Anderson (1982, 1983) processes abstract declarative knowledge with interpretative production rules, which match parts of declarative representations and create new production rules. To illustrate the flavor of the approach, consider the following didactive example (not identical to any of the author's own examples): *If you want to achieve* G, *and you know the proposition "if* S, *then* G,*" then form the new production rule: if you want to achieve* G, *then set the subgoal to achieve* S. Execution of this interpretative rule has two important consequences: It incorporates the declarative principle *if* S, *then* G into the learner's practical knowledge, and it eliminates the need to retrieve that piece of knowledge from memory. Neves and Anderson (1981) demonstrated how a collection of interpretative rules can produce executable rules for proof finding in plane geometry from declarative representations of geometry theorems.

The principles-to-actions transformation has been studied in depth in the domain of counting. Empirical studies indicate that children know very early some relevant principles, for example, that the counting words form a linear sequence, that the mapping from words to objects is supposed to be one-one, and that the last counting word represents the cardinality of the set that is counted (Gelman & Gallistel, 1978). In the COUNTPLAN model (Greeno, Riley, & Gelman, 1984; Smith, Greeno, & Vitolo, 1989), these principles are represented as action schemata, which are processed by a planning-like process to yield a plan for how to count a set of objects. A strong feature of the model is that it can generate plans for nonstandard counting tasks, for example, *count the yellow objects before the* *blue ones*. A very different process for turning the counting principles into practical knowledge is described in Ohlsson and Rees (1991a).

That these mechanisms operate in the domains of geometry and counting is no accident. The idea that action is – or ought to be – derived from principles is entrenched in mathematics education research (Hiebert, 1986). To the extent that the principled knowledge is communicated via written or spoken discourse, the problem of deriving action from abstract knowledge and of learning from instruction become intertwined. But proceduralization and planning apply equally well to knowledge retrieved from long-term memory.

3.3. *Transfer Prior Knowledge*

Initial rules for an unfamiliar task can be generated by adapting previously learned rules. That is, the problem of how practice gets under way can be subsumed under the problem of transfer of training. There are three principled ideas about how learners can utilize this source of information: identical elements, analogy, and subsumption.

3.3.1. IDENTITY

If the unfamiliar task is identical in some respects to an already familiar task, then components of the previously learned skill might apply to the unfamiliar task without change (*the identical elements hypothesis*; Thorndike, 1911, pp. 243–245). This hypothesis comes for free with a production system architecture because rules are automatically considered whenever they match the current situation. Kieras and Bovair (1986), Singley and Anderson (1989), and Pirolli and Recker (1994) report success in predicting the magnitude of transfer effects by counting the number of rules shared between two methods. However, the identical rules hypothesis predicts that positive transfer effects are necessarily symmetrical in magnitude, a dubious prediction (Ohlsson, 2007). Also, identity is a very restrictive criterion for the re-use of practical knowledge.

3.3.2. ANALOGY

The hypothesis of *analogical transfer* assumes a mapping process that identifies structural similarities between the task at hand and some already mastered task. The mapping is used to construct a method or a solution for the unfamiliar task, using the familiar one as a template. For example, consider a situation described by *Block A is on the table*, *Block B is on the table*, and *Block C is on top of Block B*. If the goal is to *put Block C on Block A*, then the successful action sequence is to *grasp* C, *lift* C *up*, *move* C *sideways*, and *put* C *down*. When the learner encounters a second situation in which *Box R is inside Box X*, *Box S is inside Box X*, *Box T is inside Box S*, and the goal is to *put T inside R*, the mapping

{*table* → *Box X*,
on top of → *inside*,
Block A → *Box R*,
etc.}

leads to the analogous solution *grasp T*, *take T out of X*, *move T sideways*, and *put T inside R*. The two analogues are not similar in any perceptual sense, but they share the same relational structure, so one can serve as a template for the other.

There are multiple ways to implement the two processes of analogical mapping and inference. The structure mapping principle proposed by Gentner (1983) and implemented in the Structure Mapping Engine (Falkenhainer, Forbus, & Gentner, 1989; Forbus, Gentner, & Law, 1994) says that higher-order relations should weigh more in choosing a mapping than lower-order relations and perceptual features. Holyoak and colleagues (Holyoak, 1985; Holyoak & Thagard, 1989a; Spellman & Holyoak, 1996) emphasized pragmatic factors, that is, which mapping seems best from the point of view of the learner's current purpose. The mapping processes by Keane, Ledgeway, and Duff (1994) and Wilson et al. (2001) are designed to minimize cognitive load, the former by satisfying a variety of constraints, for example, *map only objects* *of the same type*, and the latter by only mapping a single pair of propositions at a time. The path-mapping process proposed by Salvucci and Anderson (2001), however, pursues flexibility by separating a low-level, object-to-object mapping process from the higher-order, acquired, and hence potentially domain-specific processes that use it. Mapping processes can be implemented as connectionist networks (Holyoak & Thagard, 1989b; Hummel & Holyoak, 1997, 2003). Anderson and Thompson (1989) and Kokinov and Petrov (2001) emphasize the need to integrate analogical reasoning with other cognitive functions.

Of particular interest from the point of view of skill acquisition is the distinction between different types of analogical inferences. In some models, an analogical mapping is used to construct a solution *path* for the target problem, as in the previous didactic block/box example. Carbonell (1983, 1986; Veloso & Carbonell, 1993) proposed a *derivational analogy* mechanism of this sort. The learner infers a solution to the target problem, a sequence of actions, but no general strategy or method, so this conservative process will primarily affect behavior on the current task. In other models, an analogical mapping is used to infer a solution *method*; see VanLehn and Brown (1980) for an early attempt in terms of planning nets. At a more fine-grained level, the analogy might generate a part of a method, such as a single production rule (Anderson & Thompson, 1989; Blessing & Anderson, 1996; Pirolli, 1986, 1991). In these cases, the learner gains new practical knowledge that might apply not only to the target task but also to future tasks, a riskier type of analogical inference.

In yet another variation on the analogy theme, the EUREKA system by Jones and Langley (2005) uses analogical mapping to infer how a fully specified, past problem-solving step can be applied to the current situation. The Cascade model (VanLehn & Jones, 1993) uses a closely related mechanism. Although this application of analogy – *analogical operator retrieval* – is a part of the performance mechanism rather than a learning mechanism, it allows past steps, derivations, or problem-solving episodes, even if completely specific, to affect future behavior.

3.3.3. SUBSUMPTION

Some prior cognitive skills transfer to the target task because they are general enough to subsume the unfamiliar task at hand. The idea of wide applicability through abstraction or generality goes back to antiquity, but takes a rather different form in the context of practical as opposed to declarative knowledge. General or *weak methods* make few assumptions about the task to which they are applied, so the learner does not need to know much about the task to use them (Newell, 1990; Newell & Simon, 1972). By the same token, such methods do not provide strong guidance. Different weak methods structure search in different ways. Hill-climbing (take only steps that improve the current situation), backward search (identify what the last step before achieving the current goal would have to be and pose its requirements as subgoals, then iterate) and means–ends analysis (identify differences between the current state and the goal and think of ways to reduce each one) are the most well-known weak methods. For example, Elio and Scharf's (1990) EUREKA model initially solves physics problems via means–ends analysis, but accumulates problem-solving experiences into problem schemas that gradually come to direct future problem-solving efforts.

People also possess a repertoire of slightly more specific but still weak heuristics such as *if you want to figure out how to use an unfamiliar device, push buttons at random and see what happens*, and *if you want to know how to get to location X, ask someone*. Weak methods and heuristics are not learning mechanisms – they do not create new practical knowledge – but they serve to generate task-relevant actions. The actions produce new information about the task, which in turn can be used by a variety learning mechanisms; see the following section. When weak methods dominate initial task behavior, skill acquisition is a process of *specialization*, because it transforms those methods into domain-specific heuristics and strategies. This is a widely adopted principle (Anderson, 1987; Jones, Ritter & Wood, 2000; Langley, 1985; Ohlsson, 1996; Rosenbloom, Laird, & Newell, 1993; Sun, Slusarz, & Terry, 2005; VanLehn, 1999; VanLehn & Jones, 1993). It represents an important insight, because common sense suggests that learning proceeds in the opposite direction, from concrete actions to more abstract competencies.

There is no reason to doubt the psychological reality of either of these three transfer relations – identity, analogy, and subsumption – but there are different ways to exploit each one. Both analogy and subsumption are relaxations of the strict criterion of identity. They make prior skills more widely applicable by allowing for some differences between past and current tasks.

3.4. *Study Someone Else's Solution*

A fourth source of information on which to base initial behavior vis-à-vis an unfamiliar task is a solution provided by someone else. In an educational setting, a teacher or helpful textbook author might provide a written representation of a correct solution, a so-called *solved example*. To learn from a solved example, the learner has to study the successive steps and infer how each step was generated. There are at least three key challenges to learning from solved examples: The example might be incomplete, suppressing some (presumed obvious) steps for the sake of conciseness, which forces the learner to interpolate the missing steps. Also, a solved example might not explain why each step is the correct step where it occurs, which forces the learner to guess the correct conditions on the actions. Finally, because a solved example is specific (by definition of "example"), there is the issue how, and how far, to generalize each step.

The Sierra model (VanLehn, 1983, 1987) learned procedures from sequences of solved examples, organized into lessons, in the domain of place-value arithmetic. The examples were parsed both top-down and bottom-up. Various constraints were applied to choose a possible way to close the gap, especially the *one-subprocedure per lesson* constraint (VanLehn, 1987). Sierra produced a set of initial ("core") procedures that were not guaranteed to be complete and hence might generate impasses when executed, necessitating further learning. The main purpose of Sierra was to explain, in conjunction with Repair Theory, the origin of errors in children's arithmetic (see Section 4.2 on learning at impasses).

The Cascade model (VanLehn, 1999; VanLehn & Jones, 1993; VanLehn, Jones, & Chi, 1992) learns from solved examples in the domain of physics. The model studies examples consisting of sequences of lines. It attempts to derive each line, using its domain-specific knowledge. If the derivation succeeds, it stores the derivation itself; because Cascade uses analogies with past derivations to guide search, stored derivations can affect future processing. If the derivation fails, the system engages background knowledge that can be of various types but is likely to be overly general. If the derivation succeeds using overly general knowledge, the system applies an EBL technique called *explanation-based learning of correctness* to create a specialized version. Once it has proven its worth, the new rule is added to the learner's domain-specific knowledge. Finally, if Cascade cannot derive the line even with its general knowledge, it stores the line itself in a form that facilitates future use by analogy. Reimann, Schult, and Wichman (1993) described a closely related model of learning to solve physics problems via solved examples, using both rules and cases. The X system described by Pirolli (1986, 1991) uses analogies to solved examples to guide initial problem solving rather than overly general background knowledge, and it uses the knowledge compilation mechanism of the ACT* model rather than EBL to cache the solution for future use, but its principled approach to initial learning is similar.

In some instructional settings, it is common for a coach to *demonstrate* the correct solution, that is, to perform the task while the pupil is observing. Learning from demonstrations poses all the same problems as learning from solved examples (except possibly incompleteness), plus the problems of visual perception and learning under real-time constraints. Having to explain vision as well as learning is not a simplification, and I know of no cognitive model of human learning that learns by observing demonstrations. Learning by mimicry has played a central role in social learning theory (Bandura, 1977). Donald (1991) has made the interesting suggestion that mimicry was the first representational system to appear in hominid evolution and that remnants of it can still be seen in the play of children.

3.5. *Discussion*

The four principled answers to the question of how a learner can start practicing – follow exhortations; reason from abstract declarative knowledge; transfer identical, analogous, or general prior skills; and study someone else's solution – can be implemented in multiple ways. The diversity of approaches to the generation of initial rules is highlighted in Table 13.1. All four modes of learning have a high degree of psychological plausibility, but the validity of the exact processing details of the competing mechanisms is difficult to ascertain. Each mode is likely to produce initial rules that are incorrect or incomplete: Details might be lost in the translation of verbal recipes, reasoning can be faulty, identical elements might be incomplete, analogies might not be exact, search by weak methods might not find an optimal path, and solved examples and real-time demonstrations can be misunderstood. Exhortations, principles, prior skills, and solved examples are sources of initial rules that are likely to require fine tuning by other learning mechanisms.

4. How Are Partially Mastered Skills Improved?

The second phase of skill acquisition begins after the first correct performance and ends with mastery, that is, reliably correct performance. The learning mechanisms that are responsible for improvement during this phase answer the question, *how can an initial, incomplete, and perhaps erroneous method improve in the course of practice?* Although the mechanisms that dominate in the first phase necessarily draw on information sources available before action begins, the mechanisms that dominate this phase capitalize on the information that is generated by acting. The latter includes information to the effect that the learner is on the right track (*positive feedback*). An important subtype of positive feedback is *subgoal satisfaction*. The discovery that a subgoal has been achieved is very similar to the reception of environmental feedback in its implications for learning – the main difference is whether the information originates internally or externally – and the two will be discussed together. The environment can also produce information to the effect that an action was incorrect, inappropriate, or unproductive in some way (*negative feedback*). Feedback is both a triggering condition and a source of information, but learning from positive and negative feedback requires different processes. Another important type of triggering event is the occurrence of an *impasse*, a situation in which the learner cannot resolve what to do next.

4.1. *Positive Feedback and Subgoal Satisfaction*

As the learner acts vis-à-vis the task on the basis of initial rules, he or she will sooner or later perform a correct action or obtain some useful or desirable intermediate result. Information that designates an action or its outcome as correct or useful can originate internally (*well, that worked*), in causal consequences (*if the apple falls in your hand, you know you shook the tree hard enough*), or in utterances by an instructor (*well done*). The receipt of positive feedback is a trigger for learning. The theoretical question is what is learned. If the learner takes a correct step knowing that it is correct, there is nothing to learn. Yet, positive feedback facilitates

Table 13.1: Sample of learning mechanisms that operate in the initial phase of skill acquisition

Information source	*Name of mechanism*	*A key concept*	*Example model*	*Example domain*	*Select reference*
Instructions	Chunking	Search for a path that contains the recommended action; then contract that path into new rules.	Instructo-Soar	Blocks world	Huffman & Laird (1995)
	Construction-integration	Add instructions to a semantic network, then re-compute the distribution of activation.	ADAPT	Piloting jet airplanes	Doane & Sohn (2000)
Abstract declarative knowledge	Proceduralization	Use general interpretative rules to translate declarative principles into production rules.	ACT*	Plane geometry	Neves & Anderson (1981)
Prior practical knowledge	Identical rules	Tasks share rules, so some rules apply to a new task without change.	ACT*	Word processing	Singley & Anderson (1989)
	Structure mapping	Higher-order relations are given more weight in mapping than lower-order relations and features.	SME	Physics	Falkenhainer, Forbus, & Gentner (1989)
	Multiconstraint mapping	Select a mapping on the basis of pragmatic constraints, i.e., its relevance for the current goal.	ACME	Radiation problems	Holyoak & Thagard (1989b)
	Derivational analogy	Use analogical mapping to revise the derivation of the solution to a base problem so as to fit the target. as to fit the current task.	Prodigy	Simulated logistics	Veloso & Carbonell (1993)
Solved example provided by somebody else.	EBL of correctness	Derive the next step in a solved example, then cache the derivation into a new problem-solving rule.	Cascade	Mechanics	VanLehn (1999)

ACT* = Adaptive Cognitive Theory. SME = Structure Mapping Engine. ACME = Analogical Constraint Mapping Engine.

human learning, presumably because many steps generated by initial rules are tentative, and positive feedback reduces uncertainty about their correctness.

4.1.1. INCREASE RULE STRENGTH

The simplest mechanism for uncertainty reduction is described in the first half of Thorndike's Law of Effect (Thorndike, 1927): Increase the strength of the rule that generated the feedback-producing action. Variants of this *strengthening* idea are incorporated into a wide range of cognitive models.

The EUREKA model described by Jones and Langley (2005) stores past problem-solving steps, fully instantiated, in a semantic network memory. When faced with a decision as to what to do in a current situation S, the model spreads activation across the network to retrieve a set of past steps that are relevant for S. A step is selected for execution based on degree of similarity to the current situation. (When a problem is encountered a second time, the exact same step that led to success last time is presumably maximally similar and hence guaranteed to be selected for execution.) Finally, analogical mapping between the past step and S is used to apply the step to S. As experience accumulates, the knowledge base of past steps grows. Positive and negative feedback are used to adjust the strengths of the relevant network links, which in turn alters the outcome of future retrieval processes. In the GIPS model, Jones and VanLehn (1994) interpreted positive feedback as evidence in favor of the hypothesis that the action was the right one under the circumstances and increased the probability of that hypothesis with a probabilistic concept learning algorithm, a very different concept of strengthening.

There are multiple implementation issues: By what function is the strength increment to be computed? How is the strength increment propagated backward through the solution path if the feedback-producing outcome required N steps? How is the strength increment to be propagated upward in the goal hierarchy? Should a higher-order goal be strengthened more, less, or by the same amount as a lower-order goal (Corrigan-Halpern & Ohlsson, 2002)? Strengthening increases the probability that the feedback-producing rule will be executed in every situation in which it can, in principle, apply. But a rule that is useful in some class of situations {S} is not necessarily useful in some other class of situations {S′}. The purpose of learning must be to separate these two classes of situations, something strengthening does not accomplish.

4.1.2. CREATE A NEW RULE

Positive feedback following a tentative action A can trigger the bottom-up creation of a new rule that recommends the successful action in future encounters with the same situation. The theoretical problem is that the situation S itself is history by the time the feedback arrives and will never recur. The purpose thus cannot be to create a rule that executes A in S, but in *situations like* S. A mechanism for creating a new rule following success must provide for some level of generality.

One solution is to create a very specific rule by using the entire situation S as its condition and then rely on other learning mechanisms to generalize it. This is the solution used in the CLARION system (Sun, Merril, & Peterson, 2001; Sun et al., 2005), which is a hybrid model with both subsymbolic and symbolic learning. Actions can be chosen on the basis of a quantitative measure called a Q-value, computed by a connectionist network. When an action chosen in this way is rewarded with a positive outcome, and there is no symbolic rule that would have proposed that action in that situation, the system creates a new rule with the current state as the condition on that action. (If such a rule already exists, the rule is generalized.) The opposite solution is to create a maximally general rule and rely on other learning mechanisms to restrict its application. This solution has received less attention (but see Bhatnagar & Mostow, 1994; Ohlsson, 1987a).

The more common solution is to generalize the specific step conservatively, usually by replacing (some) constants with variables. An early model of this sort was described by Larkin (1981). It responded to successful derivations of physics quantities by creating new rules that could duplicate the derivations. Particular values of physical magnitudes were replaced with variables, on what basis was not stated. Lewis (1988) combined analogy from existing productions and explanation-based generalization to create new rules in response to positive outcomes.

Later systems have used some version of EBL to contract derivations or search paths into single rules and to provide a judicious level of generality. This principle is at the center of the Soar system (Newell, 1990; Rosenbloom et al., 1993). Soar carries out all activities through problem space search. When the goal that gave rise to a problem space is reached, Soar retrieves the search path that led to it and applies an EBL-like mechanism called *chunking* (Newell & Rosenbloom, 1981; Rosenbloom & Newell, 1986, 1987). The result is a rule of grounded generality that can regenerate the positive outcome without search. The theme of searching until you find and then using EBL or some related technique to cache the successful path with an eye toward future use recurs in several otherwise different models (e.g., VanLehn, 1999).

4.1.3. GENERALIZE RULES

When a rule already exists and generates a positive outcome, a possible response is to generalize that rule. If it is allowed to apply in more situations, it might generate more positive outcomes. In the CLARION model (Sun et al., 2001, 2005), when an action proposed by a rule generates positive feedback, the rule is generalized. Curiously, this is done by *adding* a condition element, a value on some dimension describing the current situation, to the rule. In a pattern-matching architecture, adding a condition element *restricts* the range of situations in which a rule matches, but CLARION *counts* the number of matches, so one more condition element provides one more chance of scoring a match, giving the rule more chances to apply.

If multiple rule applications and their consequences – an execution history – are stored in memory, rule generalization can be carried out inductively. In ACT*, a collection of specific rules (or rule instances) that all recommended the same action and produced positive feedback can serve as input to an inductive mechanism that extracts what the rules have in common and creates a new rule based on the common features (Anderson, 1983; Anderson, Kline, & Beasley, 1979). However, inductive, commonalities-extracting mechanisms that operate on syntactic similarities have never been shown to be powerful. Life is full of inconsequential similarities and differences, so getting to what matters usually requires analysis.

Indeed, Lenat (1983) made the intriguing observation that heuristics of intermediate generality appear to be less useful than either very specific or very general heuristics. For example, the specific heuristic, *to turn on the printer in Dr. Ohlsson's office, lean as far toward the far wall as you can, reach into the gap between the wall and the printer with your left arm, and push the button that is located toward the back of the printer*, is useful because it provides very specific guidance, whereas the general heuristic, *to turn on any electric device, push its power button*, is useful because it is so widely applicable. The intermediate heuristic, *if you want to turn on a printer, push its power button*, provides neither advantage. An inductive rule generalization mechanism is likely to produce rules of such intermediate generality.

4.2. *Interlude: Learning at Impasses*

Impasses are execution states in which the cognitive architecture cannot resolve what to do next. An impasse is a sign that the current method for the target task is incomplete in some way, so impasses should trigger learning. The mere occurrence of an impasse is not in and of itself very informative, so the question is how the inability to proceed can be turned into an opportunity to improve. The general answer is that some trick must be found that resolves the impasse and enables problem solving to continue; learning occurs when the latter produces a positive outcome. Different models differ in how they break out of the impasse as well as in how they learn from the subsequent success.

In Repair Theory (Brown & VanLehn, 1980; VanLehn, 1983, 1990), the cognitive architecture has access to a short list of *repairs*, processes it can execute when it does not know what action to take next. VanLehn (1983, p. 57) described five repairs: pass over the current step (*No-op*); return to a previous execution state and do something different (*Back-up*); give up and go to the next problem (*Quit*); revise the execution state (technically, the arguments in the top goal) so as to avoid the impasse (*Refocus*); and relax the criteria on the application of the current step (*Force*). Although applications of a repair can be saved for future use (VanLehn, 1990, pp. 43, 188), repairs are not learning mechanisms. They enable task-relevant behavior to continue and are in that respect analogous to weak methods. The purpose of Repair Theory was to explain, in combination with the Sierra model of induction from solved example, the emergence of children's incorrect subtraction procedures.

The previously mentioned Cascade model (VanLehn, 1988, 1999; VanLehn & Jones, 1993; VanLehn et al., 1992) of learning from solved examples also learns at impasses while solving physics problems. If a subgoal cannot be achieved with the system's current strategy, it brings to bear background knowledge that might be overly general. If the knowledge allows the impasse to be resolved and if a positive outcome eventually results, then a new, domain-specific rule is created using explanation-based learning of correctness. The new rule is added tentatively to the model's domain knowledge until further evidence is available as to its appropriateness or usefulness. The new domain rule is a special case of the overly general rule, so this is yet another case of specialization. If an impasse cannot be resolved even by engaging general background knowledge, the system uses a version of analogy to continue problem solving (not unlike applying a repair), but does not learn a new rule. Similarly, Pirolli's (1986, 1991) X model responded to impasses through analogies with available examples. If an analogy was successful in resolving an impasse, the resolution was stored as production rules for future use.

In the Soar system (Newell, 1990; Rosenbloom, Laird & Newell, 1993; Rosenbloom & Newell, 1986, 1987), an impasse causes the creation of a subgoal that poses the resolution of the impasse as a problem in its own right. That subgoal is pursued by searching the relevant problem space, bringing to bear whatever knowledge might be relevant and otherwise falling back on weak methods. When the search satisfies the subgoal, the problem-solving process is captured in one or more production rules through *chunking*, an EBL-like mechanism that compresses the successful search path into a single rule of appropriate generality. Another model, Icarus, which also engages in problem solving in response to an impasse, has been described by Langley and Choi (2006). This model uses a variant of backward chaining to resolve a situation in which no existing skill is sufficient to reach the current subgoal. When the solution has been found, it is stored for future use.

These models differ in how they resolve an impasse: call on repairs, apply weak methods like search and backward chaining, reason from general background knowledge, and use analogy to past problem-solving experiences. These mechanisms are not learning mechanisms; they do not change the current strategy. Their function is to allow task-oriented behavior to continue. Once the impasse is broken and problem solving continues, learning occurs at the next positive outcome via the same learning mechanisms that are used to learn from positive outcomes in general. Impasses trigger learning but do not provide unique information, so learning at impasses is a special case of learning from positive outcomes.

4.3. *Negative Feedback*

Much of the information generated by tentative actions resides in errors, failures, and undesirable outcomes. There are multiple mechanisms for making use of such information. The basic response is to avoid repeating the action that generated the negative feedback. More precisely, the problem of learning from negative feedback can be stated as follows: If rule R recommends action A in situation S, and A turns out to be incorrect, inappropriate or unproductive vis-à-vis the current goal, then what is the indicated revision of R? The objective of the revision is not so much to prevent the offending rule from executing in S, or situations like S, but to prevent it from generating similar errors in the future.

4.3.1. REDUCE STRENGTH

The simplest response to failure is described in the second half of Thorndike's Law of Effect: Decrease the strength of the feedback-producing rule. As a consequence, that rule will have a lower probability of being executed. Like strengthening, this *weakening* mechanism is a common component of cognitive models (e.g., Jones & Langley, 2005). As with strengthening, there are multiple issues: By what function should the strength values be decremented, and how should the strength decrement be propagated backward through prior steps or upward through the goal hierarchy (Corrigan-Halpern & Ohlsson, 2002)? Weakening lowers the probability that the rule will execute in any future situation. The purpose of learning from negative feedback is to discriminate between those situations in which the rule is useful from those in which it is incorrect, and a strength decrement is not an effective way to accomplish this. Jones and VanLehn (1994) instead interpreted negative feedback as evidence against the hypothesis that the action was the right one under the circumstances and reduced the probability of that hypothesis with a probabilistic concept learning algorithm.

4.3.2. SPECIALIZATION

Ohlsson (1993, 1996, 2007; see also Ohlsson et al., 1992; Ohlsson & Rees, 1991a, 1991b) has described *constraint-based rule specialization*, a mechanism for learning from a single error. It presupposes that the learner has sufficient (declarative) background knowledge, expressed in terms of constraints, to judge the outcomes of his or her actions as correct or incorrect. A constraint is a binary pair <R, C> of conditions, the first determining when the constraint is relevant and the second determining whether it is satisfied. When an action violates a constraint, that is, creates a situation in which the relevance condition is satisfied but the satisfaction condition is not, the violation is processed to create a more restricted version of the offending rule. The constraint-based rule specialization mechanism identifies the weakest set of conditions that will prevent the rule from violating the same constraint in the future. For example, if the rule is *if the goal is* G *and the situation is* S, *then do A*, and it turns out that doing A in S violated some constraint <R, C>, then the constraint-based mechanism specializes the rule by creating two new rules, one that includes the new condition *not-R* (do not recommend A when the constraint applies) and one that includes the condition C (recommend A only when the constraint is guaranteed to be satisfied); see Ohlsson and Rees (1991a) for a formal description of the algorithm. The purpose of constraint-based specialization is not primarily to prevent the rule from executing in the current situation or in situations like it, but to prevent it from violating the same constraint in the future. The algorithm is related to EBL as applied to learning from errors, but does not require the combinatorial process of constructing an explanation of the negative outcome.

The CLARION model (Sun et al., 2001, 2005) contains a different specialization mechanism: If an action is executed and followed by negative feedback, and there is a rule that proposed that action in that situation, then the application of that rule is restricted. This is done by removing a *value*, that is, a measure on some dimension used to describe the current situation. In the context of CLARION, this decreases the number of possible matches and hence restricts the range of situations in which the rule will be the strongest candidate.

A rather different conception of specialization underpins systems that respond to negative feedback by learning *critics*, rules that vote against performing in action during conflict resolution. The ability to encode missteps into critics removes the need to specialize overly general rules, because their rash proposals are weeded out during conflict resolution (Ohlsson, 1987a). This idea has been explored in machine learning research (Bhatnage & Mostow, 1994), where critics are sometimes called *censors* or *censor rules* (Bharadwaj & Jain, 1992; Jain & Bharadwaj, 1998; Winston, 1986).

The previous mechanisms improve practical knowledge by making it more specific and thereby restricting its application, in direct contrast to the idea that practical knowledge becomes generalized and more abstract over time. The latter view is common among lay people and researchers in the fields of educational and developmental psychology, in part, perhaps, as a legacy of Jean Piaget's (1950) and his followers' claim that cognitive development progresses from sensori-motor schemas to formal logical operations. "Representations are literally built from sensory-motor interactions" (Fischer, 1980, p. 481).

4.3.3. DISCRIMINATION

Restle (1955) and others tried to accommodate discrimination within the behaviorist framework, but an explanation of discrimination requires symbolic representations. Langley (1983, 1987) described SAGE, a system that included a discrimination mechanism. The model assumes that each application of a production rule, including any positive and negative feedback, and the state of the world in which the rule applied are recorded in memory. Once memory contains some instances that were followed by positive feedback and some that were followed by negative feedback, the two sets of situations can be compared with the purpose of identifying features that differentiate them, that is, that hold in the situations in which the rule generated positive outcomes but not in the situations in which it generated negative outcomes, or vice versa. One or more new rules are created by adding the discriminating features to the condition side of the original rule. A very similar mechanism was included in the 1983 version of the ACT* theory (Anderson, 1983). A rather different mechanism for making use of an execution history that records both successful and unsuccessful actions, based on quantitative concept learning methods, was incorporated into the GIPS system described by Jones and VanLehn (1994).

Implementation of a discrimination mechanism raises at least the following issues: What information should be stored for each rule application? The instantiated rule? The entire state of working memory? How many examples of negative and positive outcomes are needed before it is worth searching for discriminating features? By what criterion are the discriminating features to be identified? Which new rules are created? All possible ones? If not, then how are the new rules selected?

4.4. *Discussion*

Positive and negative outcomes are simultaneously triggers for learning and inputs to learning mechanisms. Learning from feedback is not as straightforward as Thorndike (1927) presupposed when he formulated the Law of Effect. The simplicity of early formulations hid the complexity of deciding to which class of situations the feedback refers. If doing A in S led to the attainment of goal G, what is the class {S} of situations in which A will have this happy outcome? If I see a movie by director X and lead actor Y on topic Z, and I enjoy the movie, what is the conclusion? It takes more than syntactic induction to realize that *see more movies by director X* is a more sensible conclusion than *see more movies on topic Z*. If A leads to an error, the even harder question is which revision of the responsible rule will prevent it from causing similar errors in the future. The variety of approaches to learning from feedback is highlighted in Table 13.2.

5. How Do Skills Improve Beyond Mastery?

The third phase of skill acquisition begins when the learner exhibits reliably correct performance and lasts as long as the learner keeps performing the task. During this period, the performance becomes more streamlined. Long after the error rate has moved close to zero, time to solution keeps decreasing (Crossman, 1959), possibly throughout the learner's entire lifetime, or at least until the onset of cognitive aging (Salthouse, 1996). The learning mechanisms operating during this phase are answers to the question, *how can an already mastered skill undergo further improvement?* What is changing, once the strategy for the task is correct? Even a strategy that consistently delivers correct answers or solutions might contain inefficient, redundant, or unnecessary steps. Changes of this sort are devided into three types: changes in the sequence of overt actions (optimization at the knowledge level), changes in the mental code for generating a fixed sequence of actions (optimization at the computational level), and the replacement of computation with memory retrieval. The main source of information to support these types of changes is the current *execution state* and, to the extent that past execution states are stored in memory, the *execution history* of the learner's current strategy.

5.1. *Optimization at the Knowledge Level*

The strategy a learner acquires in the course of practice might be correct but inefficient. Over time, he or she might discover or invent a shorter sequence of actions to accomplish the same task. The challenge is to explain what drives the learner to find a shorter solution when he or she cannot know ahead of time whether one exists and when there is no negative feedback (because his or her current strategy leads to correct answers).

A well-documented example of short-cut detection is the so-called SUM-to-MIN transition in the context of simple mental additions. Problems like $5 + 3 = ?$ is at a certain age solved by counting out loud, *one, two, three, four, five, six, seven, eight – so eight is the answer*. Only after considerable practice do children discover that the first five steps are unnecessary and transition to the more economical MIN-strategy, in which they choose the larger addend and count up: *five, six, seven, eight – eight.*

Neches (1987) described seven different types of optimization mechanisms in the context of the HPM model, including deleting redundant steps, replacing a subprocedure, and reordering steps, and he showed that they collectively suffice to produce the SUM-to-MIN transformation. Jones and VanLehn (1994) modeled the same shortcut discovery in their GIPS model. Each condition on a GIPS action is associated with two numerical values, *sufficiency* and *necessity*. Conflict resolution uses these values to compute the odds that the action is worth selecting, and the action with the highest odds wins. The two values are updated on the basis of successes and failures with a probabilistic concept learning algorithm. A more recent model, the Strategy Choice and Discovery Simulation (SCADS), was proposed by Shrager and Siegler (1998; see also Siegler & Araya, 2005). SCADS has limited attentional resources, so at the outset of practice, it merely executes its given strategy. Once the answers to some problems can be retrieved from memory and hence require little attention, attention is allocated to strategy change processes that (a) inspect the execution trace and delete redundant steps, and (b) evaluate the efficiency of different orders of execution of the steps in the current strategy and fixate the more efficient one (p. 408). These two change mechanisms turn out to be sufficient to discover the MIN strategy.

Another strategy shift that results in different overt behavior transforms the

Table 13.2: Sample of learning mechanisms that operate in the mastery phase of skill acquisition

Information source	*Name of mechanism*	*A key concept*	*Example model*	*Example domain*	*Select reference*
Positive feedback	Strengthening; Law of Effect, 1st part	When a positive outcome is encountered, increase the strength of the responsible action or rule.	Eureka	Tower of Hanoi	Jones & Langley (2005)
	Chunking	Search until the current goal is satisfied, then cache the path to it into a new rule.	Soar	Simulated job shop scheduling	Nerb, Ritter, & Krems (1999)
	Bottom-up creation of a new rule	If implicit choice of action leads to success, create a symbolic rule with the entire situation as its condition.	CLARION	Simulated submarine navigation	Sun, Merrill, & Peterson (2001)
	Inductive rule generalization	Extract features common to several successful rule applications and encode them as a new rule.	ACT	Categorization	Anderson, Kline, & Beasley (1979)
Negative feedback	Weakening; Law of Effect, 2nd part	When a negative outcome is encountered, decrease the strength of the responsible action or rule.	Eureka	Tower of Hanoi	Jones & Langley (2005)
	Adaptive search	Encode each unsuccessful rule application as a censor rule.	FAIL-SAFE-2	Blocks world	Bhatnagar & Mostow (1994)
	Constraint-based rule specialization	Extract new rule conditions from the mismatch between constraints and the outcomes of actions.	HS	Structural formulas in chemistry	Ohlsson (1996)
	Discrimination	Compare successful and unsuccessful rule applications; add discriminating features to rule.	SAGE	Balance scale task	Langley (1987)

CLARION = Connectionist Learning with Adaptive Rule Induction ONline. ACT = Adaptive Cognitive Theory. HS = Heuristic Self-Improvement. SAGE = Strategic Acquisition Governed by Experimentation.

novice's laborious problem solving through means–ends analysis or backward chaining into the expert's forward-inference process that develops the knowledge about a problem until the desired answer can be found, perhaps without ever setting any subgoals. The ABLE model of physics problem solving by Larkin (1981) simulated this transformation in the domain of physics. Elio and Scharf (1990) achieved the same effect, also in the domain of physics, with sophisticated indexing of successful problem-solving episodes in memory. Their EUREKA model created problem solving schemas and used positive and negative outcomes to adjust the level of generality of the schemas. Over time, it relied increasingly on the forward-inference schemas and less on means–ends analysis.

Anderson (1982, 1983) explained both the transition from backward to forward chaining and the transition from serial to parallel search in the Sternberg short-term memory task by showing that rule composition can squeeze subgoals out of rules. In contrast, Koedinger and Anderson (1990) attributed the forward-inference behavior of geometry experts to a repertoire of diagram chunks that allow experts to quickly identify possible inferences in a geometric diagram, thus seemingly arriving at conclusions before they derive them, but Koedinger and Anderson did not model the acquisition of those diagram chunks. Taking a different tack, Blessing and Anderson (1996) argued that rule-level analogies suffice to discover strategic shortcuts.

Another empirically documented strategy discovery is the invention of the pyramid recursion strategy of Tower of Hanoi. Unlike the MIN-to-SUM and backward-to-forward transitions, the transition from moving single discs to moving pyramids of discs requires an *increase* in the complexity of internal processing to simplify overt behavior. Ruiz and Newell (1993) modeled this strategy discovery in the Soar system by adding special productions that (a) notice subpyramids and (b) reason about spatial arrangements like stacks of objects, but without postulating any other learning mechanisms than Soar's standard impasse-driven chunking mechanism.

A different hypothesis about shortcut detection is that the mind reasons from declarative background knowledge to new production rules that may represent shortcuts (Ohlsson, 1987b). For example, if the current strategy contains a production rule that matches goal G and produces some partial result B, and there is in memory a general implication *A_1 and A_2 implies B*, then it makes sense to create the new rule, *if you want G and you have A_1, set the subgoal to get A_2*, as well as, *if you want G and you have both A_1 and A_2, infer B*. The first rule encodes a backward-chaining subgoaling step – get the prerequisites for the target conclusion – and the second new rule is akin to the result of the proceduralization process discussed previously. This and two other mechanisms for reasoning about a set of rules on the basis of general *if–then* propositions were implemented in a model called PSS3, which reduced the simulated time for performing a simple spatial reasoning task by two orders of magnitude.

Several of these learning mechanisms require that production rules can test for properties of other production rules – the mental code, not merely traces of executions – a psychologically problematic assumption. Although these mechanisms are intended to explain success in strategy revision, Fu and Gray (2004) provide a useful counterpoint by specifying some of the conditions and factors that might prevent optimization mechanisms from operating and hence keep the performer on a stable but suboptimal solution path.

5.2. *Optimization at the Computational Level*

Even when the learner cannot find a shorter action sequence, he or she might be able to save on the mental computations required to generate the relevant sequence. In this case, overt behavior does not change, but the learner produces that behavior with fewer or less capacity-demanding cognitive steps.

An optimization mechanism, *rule composition*, was invented by Lewis (1987) and included in the ACT* model (Anderson, 1983; Neves & Anderson, 1981). This mechanism requires a less extensive access to the execution trace than shortcut detection: It need only keep track of the temporal sequence of rule executions. If two rules are repeatedly executed in sequence, then a new rule is created that performs the same work as the two rules. To illustrate the flavor of this type of change, imagine that $G, S_1 \rightarrow A_1$ and $G, S_2 \rightarrow A_2$ are two rules that repeatedly execute in sequence. A plausible new rule would be $G, S_1 \rightarrow A_1; A_2$, which is executed in a single production system cycle. Given that A_2 is always performed after A_1, there is no need to evaluate the state of the world after A_1. A full specification of this contraction mechanism needs to take interactions between the action of the first rule and the conditions of the second rule into account. In ACT*, composition worked in concert with proceduralization. The combination of the two mechanisms was referred to as *knowledge compilation*.

Recently, the composition mechanism has been replaced by the related *production compilation* mechanism (Taatgen, 2002, 2005; Taatgen & Anderson, 2002; Taatgen & Lee, 2003). The triggering condition for this learning mechanism is also that two rules repeatedly execute in temporal sequence, and, again like composition, it creates a single new rule. The resulting rule is specialized by incorporating the results of retrievals of declarative information into the resulting rule. The combination process eliminates memory retrieval requests in the first rule and tests on retrieved elements in the second rule. For example, if the two rules *if calling X, then retrieve his area code* and *if calling X and his area code is remembered to be Y, then dial Y* are executed in the course of calling a guy called John with area code 412, production compilation will create the new rule *if calling John, then dial 412*. Because there can only be a single request on memory in any one ACT-R production rule, eliminating such requests saves production system cycles.

However, there is more to combining rules than mere speed-up. Anderson (1986) argued that knowledge compilation can mimic the effects of other learning mechanisms, such as discrimination and generalization, and produce qualitatively new practical knowledge. In the same vain, Taatgen and Anderson (2002) modeled the transition from incorrect use of regular past tense for irregular verbs like "break" to the correct irregular form, using nothing but production compilation. The effects of optimization by contraction are more complicated than they first appear and deserve further study.

The issues in designing a rule combination mechanism include: What is the triggering criterion? How many times do the two rules have to execute in sequence for there to be sufficient reason to compose them? Does the new rule replace the previous rules or is it added to them? Are there counterindications? If the learner's execution history for the relevant rules *also* contains situations in which the two rules did not execute in sequence, should the rules nevertheless be combined?

5.3. *Retrieve Solutions from Memory*

If people perform the same tasks over and over again, they eventually remember the answers and hence need not perform any other processing than retrieving those answers from long-term memory. In a restricted domain such as arithmetic, the balance between computing and retrieving might over time shift in favor of retrieval. A shift from, for example, 60% of answers being calculated and 40% retrieved to the opposite percentages might have a strong effect on the mean solution time.

This shift toward memory-based responding is central to the instance-based model by Logan (1998) and the series of models of children's strategy choices in arithmetic described by R. S. Siegler and associates: the distribution of associations model (Siegler & Shrager, 1984); the Adaptive Strategy Choice Model (ASCM; Siegler & Shipley, 1995); and the SCADS model (Shrager & Siegler, 1998). All three models use the idea that associations between particular problems and their answers are gradually strengthened until they can provide a solid basis for answering. Hence, the proportion of memory-based responses increases with practice.

The psychological reality of instance memorization and a gradual shift toward memory-based responding as experience of a task domain accumulates is hardly in doubt (everyone knows the multiplication table), but this type of learning cannot be important in all domains. For example, it does not apply to buying a house because few people buy the same house multiple times.

5.4. *Discussion*

Cognitive psychologists discuss the long-term consequences of practice in terms of two concepts that in certain respects are each others' opposites: *automaticity* and *expertise*. The essential characteristics of automaticity include rigidity in execution and a high probability of being triggered when the relevant stimuli are present (Schneider & Chein, 2003). The consequences include capture errors (Reason, 1990), *Einstellung* effects (Luchins & Luchins, 1959), and negative transfer (Woltz, Gardner & Bell, 2000). But we think of experts as exhibiting a high degree of awareness, flexibility, and ability to adapt to novel situations (Ericsson et al., 2006). Which view is correct? If one practices four hours a day, six days a week, for ten years, does one end up a rigid robot or an elastic expert? Both end states are well documented, so the question is which factors determine which end state will be realized in any one case. Ericsson, Krampe, and Tesch-Rober (1993) have proposed that experts engage in deliberate practice, but they have not offered a computational model of how deliberate practice might differ from mere repetitive activity in terms of the cognitive processes involved. Deliberate practice is undertaken with the intent to improve, but how does that affect the operation of the relevant learning mechanisms? Salomon and Perkins (1989) has summarized studies that indicate that the variability of practice is the key, with more variability creating more flexible skills. Another hypothesis, popular among educational researchers, is that flexibility is a side effect of conceptual understanding. To explain the difference between automaticity and expertise, a model cannot postulate two sets of learning mechanisms, one that produces rigidity and one that leads to flexibility. The theoretical challenge is to show how one and the same learning mechanism (or set of mechanisms) can produce either automaticity or expertise, depending on the properties of the training problems (complexity, variability, etc.), the learner, the learning scenario, or other factors. It is not clear whether the current repertoire of optimization mechanisms (see Table 13.3 for an overview) is sufficient in this respect.

6. Capture the Statistical Structure of the Environment

As the learner becomes familiar with a particular task environment, he or she accumulates information about its quantitative and statistical properties. For example, the members of a tribe of foraging hunter-gatherers might have implicit but nevertheless accurate estimates of the average distance between food sources and the probability of discovering a new food source in a given amount of time, for example, before the sun sets or before winter sets in (Simon, 1956). Quantitative information of this sort was abundant in the environments in which human beings evolved (*How often has such and such an animal been sighted recently? How many days of rain in a row should we expect? How high up the banks will the river flood?*), so it is plausible that they evolved cognitive mechanisms to capture it. A modern descendant might use such mechanisms to estimate the expected travel time to the airport or the probability that a sports team will win its next match.

The behaviorist learning theories of the 1895–1945 era were the first psychological theories to focus on the effect of environmental quantities, especially the frequency,

Table 13.3: Sample of learning mechnisms that operate in the optimization phase of skill acquisition

Information source	*Name of mechanism*	*A key concept*	*Example model*	*Strategy transition*	*Select reference*
Execution trace	Redundancy elimination	The execution trace is inspected for redundant steps.	HPM	Arithmetic: SUM-to-MIN	Neches (1987)
	Reordering	Steps are ordered in the most efficient sequence.	SCADS	Arithmetic: SUM-to-MIN	Shrager & Siegler (1998)
	Chunking	If production rules notice complex situation features, chunking will incorporate them into rules.	Soar	Tower of Hanoi: disk-to-pyramid	Ruiz & Newell (1993)
	Rational learning	Use general implications to deduce a short cut; create rules that compute the shorter path.	PSS3	Reasoning: center-to-periphery	Ohlsson (1987b)
	Indexing	Solutions are accumulated in memory and indexed with abstract problem features.	EUREKA	Physics: backward-to-forward	Elio & Scharf (1990)
Temporal sequence of rule executions	Composition	Two production rules that fire in sequence are contracted into a single rule.	ACT*	Sternberg: serial-to-parallel	Anderson (1982)
	Production compilation	Two production rules that fire in sequence are contracted into a single rule.	ACT-R	Vocabulary: regular-to-irregular	Taatgen & Anderson (2002)
Memory of past answers	Association strengthening	Problem-answer associations grow stronger with use, so more answers are retrieved rather than computed.	ACSM	Arithmetic: computation-to-retrieval	Siegler & Shipley (1995)

Each example simulates some empirically documented strategy transition. HPM = Heuristic Procedure Modification. SCADS = Strategy Choice and Discovery Simulation. PSS3 = Production System Stockholm 3. ACT = Adaptive Cognitive Theory. ACSM = Adaptive Strategy Choice Model.

type, and amount of feedback (also known as reinforcement), on skill acquisition. The first theories of this sort were proposed by E. Thorndike, E. R. Guthrie, C. L. Hull, E. C. Tolman, B. F. Skinner, and others; Hilgard and Bower (1966) wrote the classical review. These theorists conceptualized the effect of feedback in cause–effect and motivational terms: Each event impacts the learner, and the effect of multiple events is merely the sum of their impacts. The strength of the disposition to perform an action could not yet be seen as an estimate of the relative frequency of environmental events like positive and negative feedback, because the learner was not yet seen as an information processor.

Mathematical psychologists in the 1945–1975 period discovered and investigated several types of adaptation to quantitative properties of the environment (see, e.g., Neimark & Estes, 1967). In a standard laboratory paradigm called *probability matching*, subjects are presented with a long sequence of binary choices (e.g., left, right) and given right–wrong feedback on each. The relative frequencies of trials on which "left" or "right" is the correct response is varied between groups. Over time, the relative frequencies of the subjects' responses begin to match the relative frequencies of the feedback, so if "left" is the correct response 80% of the time, then the subject tends to say "left" 80% of the time. In the absence of other sources of information, probability matching provides a lower hit rate than choosing the response that is more often followed by positive feedback on every trial. Other well-documented sensitivities to event frequencies include word frequency effects, prototype effects in classification, the impact of co-occurrences on causal reasoning, the role of estimated outcome probabilities in decision making, and many more.

Two distinct lines of research laid the foundation for the contemporary concept of *implicit* or *subsymbolic* learning (see Chapter 14 in this volume). Models of *semantic networks* (Anderson & Pirolli, 1984; Collins & Loftus, 1975; see Chapter 8 in this volume) contributed the important idea of splitting the quantity associated with a knowledge unit into two. The *strength* of a unit is an estimate of its past usefulness. It moves up or down according to the feedback generated when the unit is active. *Activation* is a transient quantity that estimates the moment-to-moment relevance for the situation at hand. When activation spreads across the network, the amount of activation each unit receives from other nodes is proportional to its strength. Models of *neural networks*, also known as connectionist models, have extended this basic idea with mathematically sophisticated studies of different schemes and regimens for the propagation of strength adjustments throughout a network (see Chapter 2 in this volume).

Although connectionist and symbolic network models were initially conceived as competing accounts of human learning, modelers have come to realize that it is more fruitful to see them as complementary. Schneider and Oliver (1991) and Schneider and Chein (2003) described CAP2, a hybrid model that could transfer information from the symbolic to the subsymbolic level, thus speeding up learning at the notoriously slow connectionist level. In the CLARION model (Sun et al., 2001, 2005), a connectionist network is used to represent actions and to select an action to perform in a particular situation. A combination of two connectionist learning algorithms are used to adjust the network to experience. The distributed representation models implicit skill. CLARION extracts explicit rules from such implicit knowledge and generalizes and specializes them in response to positive and negative feedback, thus relating the symbolic and subsymbolic levels in the opposite way compared with CAP2. In yet another twist on this two-roads-to-mastery theme, the Soar architecture has been revised to augment its symbolic chunking mechanism with mechanisms for so-called reinforcement learning, that is, the learning of a quantitative function for predicting the value of executing a particular action in a given situation (Nason & Laird, 2005). In this case, symbolic and statistical learning mechanisms are conceptualized to run in parallel and to be mutually supporting, without either level dictating what should be learned at the other.

In a series of path-breaking analyses, Anderson (1989, 1990) has developed the idea that mental quantities, like strengths, are estimates of environmental magnitudes, not repositories of causal impacts, into a radical new approach to the modeling of learning. The starting point for his *rational analysis* is that the mind is maximally efficient, that is, it solves each information processing task as well as the nature of the task allows. Consequently, the structure of the mind mirrors the structure of the task environment, a fundamental point often illustrated with a mythical beast called Simon's ant; its winding path across a beach reflects the topology of the beach more than its decision-making mechanism (Simon, 1970). In humans, the task of the long-term memory system is to correctly estimate, for each memory entry, the probability that it is the entry that needs to be retrieved in the next unit of time, given the person's current situation and goal. This probability can be estimated from prior experience: How often has the entry been needed in the past, how much time has gone by since it was needed last, and how does the probability that an entry is needed depend on the time since it was needed last? The memory entry to be retrieved is the one with the highest estimated probability of being the one that is needed. When the outcome of the retrieval becomes available – did the retrieved entry support goal attainment? – the relevant probabilities can be updated with Bayes' rule, a sound statistical inference rule (see Chapter 3 in this volume).

When a rational analysis is integrated into a cognitive architecture, the combination extends cognitive modeling in multiple ways. First, seeing mental quantities as estimates of environmental quantities focuses modelers' attention on the need for a serious analysis of the latter; which environmental quantities are people sensitive to, and how do those quantities, in fact, behave? Astonishing regularities have been found. The probability that an email address is the next one to be needed declines over time according to the same function as the probability that a particular word is the next one to appear in a newspaper headline (Anderson & Schooler, 1991, p. 401). To make accurate estimates, our brains must be sensitive to the shape of such functions. To make accurate models, modelers have to identify those functions by studying the environment, an unfamiliar type of activity. Second, the rationality assumption and the use of Bayes' rule and other sound inference rules emphasizes the question of how closely the operation of the mind approximates the maximally possible performance. If the approximation is close, then behavior can be predicted by asking how a maximally efficient system would behave. It turns out that at least some behavioral regularities, including forgetting curves, can indeed be predicted this way (Anderson & Schooler, 1991) without any processing assumptions.

The theory and practice of rational analysis is a growing enterprise (Anderson & Lebiere, 1998; Oaksford & Chater, 1998; Petrov & Anderson, 2005; Schooler & Hertwig, 2005), Indeed, the practice of modeling adaptation as a process of adjusting cognitive magnitudes so as to estimate environmental magnitudes is, in general, a growing enterprise, and it has been applied to many types of mental quantities (e.g., Altmann & Burns, 2005; Gray, Shoelles, & Sims, 2005).

How do mental estimates of environmental magnitudes help optimize a cognitive skill in the long run? Consider the following everyday example: Many of the operations I perform during word processing cause a dialogue window to appear with a request for confirmation of the operation; for example, do I really intend to shut down my computer, print this file, and so forth. After using the same computer and the same software for several years, I know exactly where on my computer screen the dialogue box and hence the confirmation button will appear. Before my computer presents the dialogue box, I have already moved my cursor to that position, so there is zero time lag between the appearance of the button and the click (see Gray & Boehm-Davis, 2000, for other examples of such micro-strategies). This rather extreme adaptation to the task environment is a case of computational optimization (clicking fast and clicking slow are equally correct), and it depends crucially on having sufficient experience for the estimate of the button location to become stable and accurate. Other quantities affect processing in other ways, optimizing memory retrieval, conflict resolution, goal setting, attention allocation, and so on. As practice progresses, the internal estimates of the relevant environmental quantities become more accurate and less noisy, and enable fine tuning of the relevant processes. Capturing the statistical structure of task environment is likely to be responsible for a significant proportion of the speed-up that accompanies practice in the long run.

7. Obstacles and Paths to Further Progress

Research on skill acquisition draws on a century of scientific progress. The computational models proposed since 1979 address a wide range of theoretical questions, and they are more detailed, more precise, and more explanatory than the verbal formulations and mathematical equations that preceded them. Nevertheless, there are reasons to despair.

What is the next step? According to the textbook definition of research, the next wave of skill acquisition research should consist of empirical studies to test which of the many hypotheses previously reviewed is correct. Researchers are supposed to derive the behavioral predictions from each hypothesized learning mechanism and test them against empirical data. The hypotheses that remain standing at the end of the day are the true ones.

There are several problems with this cartoon of research. The most serious for present purposes is the implicit assumption that we can test hypothesized learning mechanisms one at a time. But it is unlikely that the human capacity for change relies on a single learning mechanism, so we should not expect a single learning mechanism to explain all behavioral phenomena associated with skill acquisition. Instead, we should expect the theory we seek to include a repertoire of N interacting learning mechanisms (Anderson, 1983; Gagné, 1970). The total amount of behavioral change that occurs in each successive phase of training is divided among the members of that repertoire, and the relative contributions of the different mechanisms shift gradually as practice continues. Figure 13.2 shows a didactic example. The relative contributions of four sources of information are plotted in terms of the proportion of the total behavioral change within each phase that each source accounts for, in a hypothetical learner who has a single learning mechanism for each source. The character of each phase is determined by the relative contributions of different types of information and associated learning mechanisms, and these contributions shift across time. In the figure, instructions and solved examples account for almost all change in the initial phase, whereas responding to feedback is the most important type of learning in the middle phase and memory-based responding becomes dominant in the optimization phase. This is a hypothetical, didactic case. Each repertoire of learning mechanisms will exhibit its own succession of shifts.

The methodological difficulty is that we have no way of disabling N-1 of the mechanisms to test different models of the Nth one. Instead, all N mechanisms must be assumed to be operating continuously and in parallel. If so, then the overt, observable change in behavior is a product of the interactions among the N learning mechanisms. Behavioral data cannot speak to the existence or nature of any one mechanism but only to some ensemble of mechanisms. Hence, we cannot test hypotheses about, for example, analogical transfer per se, only about analogical transfer in the context of whichever other learning mechanisms we believe that people possess. Attempts to derive a specific phenomenon like the power law of practice from a single learning mechanism (a widespread misdemeanor; see, e.g., Anderson, 1982; Logan, 1998; Newell & Rosenbloom, 1981; Ohlsson,

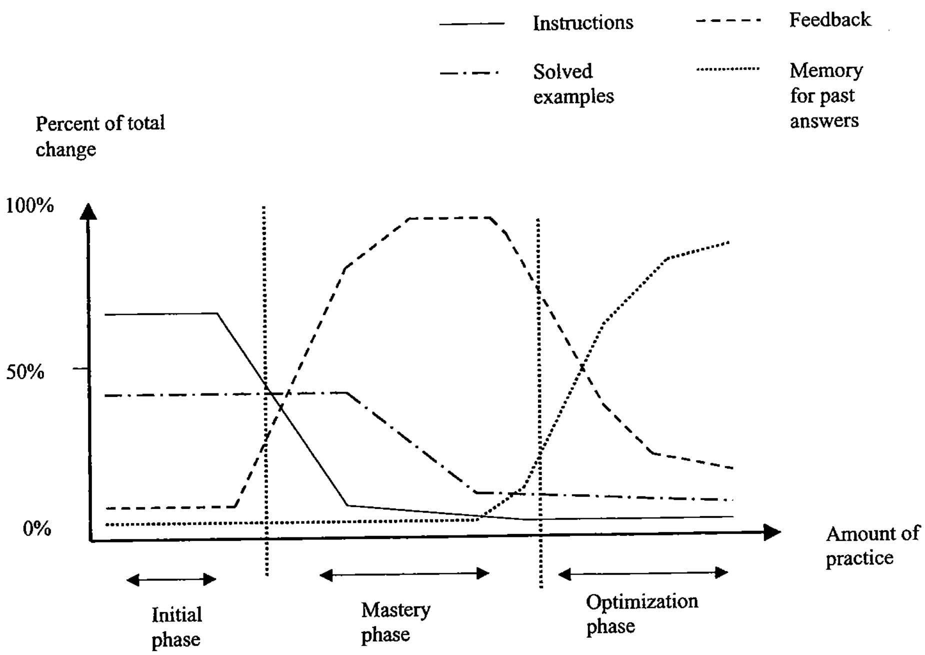

Figure 13.2. Successive shift in the relative importance of four sources of information across the three phases of skill practice in a hypothetical learner. Instruction = task instructions, exhortations, and so forth. Examples = solved examples and demonstrations. Feedback = positive and negative on-line feedback from the environment during practice. Memory = accumulated store of information about the task, including past solutions and answers, execution history of the skill, and cumulative statistics about the task environment.

1996; Shrager, Hogg, & Huberman, 1988) are unwarranted, and the outcomes of such single-mechanism derivations are uninterpretable (Ohlsson & Jewett, 1997).

Pursuing the interaction argument to its natural end point, we reach a radically counterintuitive conclusion: If a single learning mechanism successfully predicts a particular behavioral phenomenon, the hypothesis it represents must be false. Furthermore, the better the fit between the hypothesis and the data, the stronger the reason to reject the hypothesis. These conclusions follow because the behavior that results from the interactions among N learning mechanisms is highly likely to differ from the behavior that would result if the mind only possessed a single mechanism. Hence, if a single-mechanism hypothesis accounts for certain data *in the absence of interactions with other learning mechanisms*, then it is highly *unlikely* that it will fit those same data in the presence of such interactions. In other words, fit to data in a single-mechanism simulation is good reason to doubt that good fit would result if the interactions with other mechanisms were factored into the generation of the predictions. Because we have stronger reasons for believing in the existence of multiple learning mechanisms than for believing in any one mechanism, the rational response to a good fit by a model built around a single learning mechanism is to reject that model.

This methodological conundrum prevents modelers from carrying out the divide-and-test program of cartoon science. Modelers have to investigate and test entire repertoires of learning mechanisms. The resulting theory space is immense. For example, with ten different potential learning mechanisms, each of which can be either included or excluded, there are 2^{10}, or 1,024 different models to test. If it takes 10 years to evaluate each one, we will know the true theory in the spring of the year 12,246 A.D.

The standard response to a large search space is to apply information to prune the possibilities. However, the interaction argument applies to falsification as well as confirmation: If a single-mechanism model fails to fit a data set, nothing follows. That mechanism might account for those data very well in interaction with other mechanisms (Ohlsson & Jewett, 1997). It is not obvious how cognitive modelers are to claw, peck, or push their way out of this problem box.

One approach is to adopt the spirit of rational analysis. Human beings are such good learners – we are not merely superior, but orders of magnitude superior to other animals in this respect – that we might in fact be maximally efficient learners. Once prehumans adopted the evolutionary strategy of relying on learned rather than innate competencies, selective pressures propelled the evolution of multiple distinct learning mechanisms, one for every type of information that might be available to the learner during skill practice. Each mechanism is designed to make maximal use of the type of information it takes as input, and, in conjunction, the mechanisms cover all potentially available sources of information. As the retinas in our eyes are maximally sensitive in the sense that they can react to a single photon, so the learning mechanisms in our head are collectively sufficient to make use of every bit of information. I call this rationality principle the *Principle of Maximally Efficient Learning.*

To evaluate the plausibility of this proposal, the reader might try the thought experiment of imagining a model of skill acquisition that lacks one or more modes of learning reviewed in the previous sections. Which one might turn out false if we could make the appropriate empirical test? Is it plausible that we would ever conclude that people cannot benefit from instructions and exhortations? That they cannot re-use prior skills? Are unable to capitalize on positive outcomes produced by tentative steps? Cannot learn from their errors? Have no way of responding to impasses? Cannot find shortcuts or optimize their mental code? Do not absorb the statistical structure of the environment? The idea that we would ever reject any one of these hypotheses on the basis of data is quixotic, because the intuition that people can learn in each and every one of these ways is stronger than our belief in any one experimental study.

The previous sections identified nine distinct sources of information for learning: exhortations, abstract declarative knowledge, prior skills, solved examples and demonstrations, positive feedback (including subgoal satisfaction), negative feedback and errors, execution states and histories, memory for problem-answer links, and the statistical properties of the environment. Recognition of the fact that these different sources of information are available to a learner is by itself a conceptual advance in the study of skill acquisition. If we assume that people can use each type of information and that different processes are needed to use each type, as per the Information Specificity Principle, then a first-approximation model of skill acquisition should be equipped with at least nine different learning mechanisms, one for each source of information.

The first mission for such a model might be to simulate the broad functionality of human learning. The model should be able to acquire the entire range of cognitive skills that people can learn (universality); draw on information from diverse sources during learning (integration); be able to make progress even when insufficient information is available (graceful degradation); override prior experience when what is learned turns out to be a mistake or the task environment changes (defeasibility); and accumulate learning effects over long periods of practice (accumulation). Once a set of learning mechanisms has been shown to capture these and other broad features of human learning, quantitative data can be brought to bear to identify the exact details of each mechanism. If the model has N learning mechanisms, the implementation of each mechanism can be seen as a parameter. By varying the implementations, we might be able to improve the model's fit to empirical phenomena. This *functionality first, fit later* research strategy requires complex models, but it takes the logic of the interaction argument seriously.

A breakthrough in our understanding of skill acquisition can be sought along two other lines of inquiry. The set of possible or plausible learning mechanisms is constrained by the structure of the knowledge representations they operate on. Different representations afford or suggest different types of changes. One reason the production rule format has attracted skill acquisition researchers is that the rule notation is fertile in such suggestions: The ideas of adding or deleting conditions elements or of composing rules are irresistibly suggested by the notation itself. Likewise, nodes and links in a semantic network afford the ideas of creating and deleting links, whereas propositional representations afford the replacement of constants with variables (or vice versa). Although a deeper understanding of the space of possible knowledge representations was heralded in the early years of cognitive science as one of the field's major goals and potentially unique contributions, the problem of identifying exactly how the mind encodes information turned out to be intractable. For example, debates about the different roles of phonetic and semantic codes in short-term memory (Baddeley & Levy, 1971; Kintsch & Buschke, 1969; Wickelgren, 1969) and about holistic versus propositional encodings of images (Anderson, 1978, 1979; Pylyshyn, 1979) were intensively engaged but put aside without resolution. By the 1990s, the field had settled for an irregular collection of partial solutions, including semantic networks, sets of propositions, production rules, and schemas (Markman, 1999). Each of these representations captures some but not all features of human knowledge, and collectively they have no principled rationale over and above the fact that we know how to implement them in computer code.

A second wave of inquiry into knowledge representation might provide us with a new and psychologically accurate representation which, in turn, will suggest the right set of learning mechanisms. Because the individual concept is the building block for any knowledge representation, such an enterprise might benefit from linking skill acquisition research to theories of semantics. If we could understand the mental representation of a single concept (e.g., *make tea*), perhaps we could understand how the mind links concepts into the larger structures that make up practical knowledge. Whether such a project would benefit most from reaching back to the semantic theories proposed in decades past (Jackendoff, 1983; Lakoff, 1971; Miller & Johnson-Laird, 1976), attending to more recent semantic endeavors (e.g., Engelberg, 2004; Pustejovsky, 2005), or striking out in an entirely new direction (e.g., Fauconnier & Turner, 2002; Gärdenfors. 2000) is an open question, but a grounded theory of representation might constrain the set of plausible learning mechanisms.

A similar situation holds with respect to motor action. The field of computational modeling has settled on the so-called STRIPS operator as the standard representation for actions (Fikes & Nilsson, 1993), but this representation bears little relation to motor schemas and other types of representations that are discussed in the neighboring and yet so distant field of motor skill learning (Adams, 1987; Fischer, 1980; Gallistel, 1980; Wolpert, Ghahramani & Flanagan, 2001; see Chapter 24 in this volume). Again, an empirically grounded theory of the representation of elementary actions might suggest novel learning mechanisms or help modelers to choose among those already under consideration.

Another promising but as yet untapped source of constraints on the repertoire of learning mechanisms is the brain. There is no evidence that the mind is a blob of ectoplasm hovering somewhere just out of sight from scientific observation, so it must be assumed that every learning mechanism will one day be understood as implemented in neural matter. Cognitive descriptions of processes in the mind are functional descriptions of what this or that piece of wetware is doing, what function it carries out. This perspective points to the need to understand the relation between learning mechanisms like those reviewed in this chapter and modes of neural plasticity. How does the neural matter change when a learner changes his or her mind in some particular way, for example, by collapsing two production rules into one, creating a new subgoal, or lowering the strength of a link?

How many distinct modes of neural plasticity are there? One or one hundred? The cognitive neuroscience literature indicates that the answer is somewhere in between. There is *long-term potentiation* (the lowering of the threshold for signal transmission between one neuron and a downstream neuron consequent on repeated transmissions), *consolidation*, and *synaptic pruning*, but also the creation of new synapses and even wholesale replacement of brain cells. The challenge is to understand how these changes in neural matter relate to changes of mind as described at the cognitive level. For example, is long-term potentiation the same as strengthening? That is, can we reasonably assume that long-term potentiation is operating each time one of our simulation models upgrades the strength of a link as a function of frequency of use (Martin, Grimwood, & Morris, 2000; Martin & Morris, 2002)? Or is it not possible to map cognitive change mechanisms onto modes of neural plasticity in this way? Once we are past the potentiation-strengthening linkage, it becomes suspiciously difficult to make further mappings of this sort. For example, which type of neural plasticity should we conceive as implementing discrimination learning? Perhaps synaptic pruning can serve that purpose? But is pruning sensitivity to feedback? So little is understood about how to make such mind-brain mappings that it is unclear whether this approach is a tool or a problem, but it seems likely that an exhaustive list of the modes of neural plasticity would have implications for the plausibility of rival learning mechanisms defined at the cognitive level.

In the end, what is needed for further progress is a brilliant guess. Somebody has to propose a repertoire of learning mechanisms that happens to be close enough to the truth so that incremental improvement of fit to data can be pursued through the hypothesis-testing and parameter-fitting procedures of normal science. Cognitive skill acquisition awaits its Newton.

Acknowledgments

The preparation of this chapter was supported by Grant No. N00014-00-1-0640 from the Cognitive Science Program of the Office of Naval Research (ONR) to Barbara Di Eugenio and Stellan Ohlsson. The opinions expressed are those of the author. No endorsement should be inferred.

References

Ackerman, P. L. (1990). A correlational analysis of skill specificity: Learning, abilities, and individual differences. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 16*, 883–901.

Adams, J. A. (1987). Historical review and appraisal of research on the learning, retention, and transfer of human motor skills. *Psychological Bulletin, 101*, 41–74.

Altmann, E. M., & Burns, B. D. (2005). Streak biases in decision making: Data and a memory model. *Cognitive Systems Research*, 6, 5–16.

Amir, E., & Maynard-Zhang, P. (2004). Logic-based subsumption architecture. *Artificial Intelligence, 153*, 167–237.

Anderson, J. R. (1978). Arguments concerning representations for mental imagery. *Psychological Review, 85*, 249–277.

Anderson, J. R. (1979). Further arguments concerning representations for mental imagery: A response to Hayes-Roth and Pylyshyn. *Psychological Review, 86*, 395–406.

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

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

Anderson, J. R. (1986). Knowledge compilation: The general learning mechanism. In R. S. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach* (Vol. 2, pp. 289–310). Los Altos, CA: Kaufmann.

Anderson, J. R. (1987). Skill acquisition: Compilation of weak-method problem solutions. *Psychological Review, 94*, 192–210.

Anderson, J. R. (1989). A rational analysis of human memory. In H. L. Roediger III & F. I. M. Craik (Eds.), *Varieties of memory and consciousness* (pp. 195–210). Hillsdale, NJ: Lawrence Erlbaum.

Anderson, J. R. (1990). *The adaptive character of thought.* Hillsdale, NJ: Lawrence Erlbaum.

Anderson, J. R. (1993). *Rules of the mind.* Hillsdale, NJ: Lawrence Erlbaum.

Anderson, J. R., Kline, P., & Beasley, C. M. (1978, February). *A theory of the acquisition of cognitive skills* (Technical Report 77-1). New Haven, CT: Yale University.

Anderson, J. R., Kline, P. J., & Beasley, C. M. (1979). A general learning theory and its application to schema abstraction. In G. H. Bower (Ed.), *The psychology of learning and motivation: Advances in research and theory* (Vol. 13, pp. 277–318). New York: Academic Press.

Anderson, J. R., & Lebiere, C. (1998). *The atomic components of thought.* Mahwah, NJ: Lawrence Erlbaum.

Anderson, J. R., & Pirolli, P. L. (1984). Spread of activation. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 10,* 791–798.

Anderson, J. R., & Schooler, L. J. (1991). Reflections of the environment in memory. *Psychological Science, 2,* 396–408.

Anderson, J. R., & Thompson, R. (1989). Use of analogy in a production system architecture. In S. Vosniadou & A. Ortony (Eds.), *Similarity and analogical reasoning* (pp. 267–297). Cambridge, UK: Cambridge University Press.

Anzai, Y., & Simon, H. A. (1979). The theory of learning by doing. *Psychological Review, 86,* 124–140.

Baddeley, A. D., & Levy, B. A. (1971). Semantic coding and short-term memory. *Journal of Experimental Psychology, 89,* 132–136.

Bandura, A. (1977). *Social learning theory.* Upper Saddle River, NJ: Prentice Hall.

Bharadwaj, K. K., & Jain, N. K. (1992). Hierarchical censored production rule (HCPRs) system. *Data & Knowledge Engineering, 8,* 19–34.

Bhatnagar, N., & Mostow, J. (1994). On-line learning from search failure. *Machine Learning, 15,* 69–117.

Blessing, S. B., & Anderson, J. R. (1996). How people learn to skip steps. *Journal of Experimental Psychology: Learning, Memory, & Cognition, 22,* 576–598. [Reprinted in T. A. Polk & C. M. Seifert (Eds.). (2002). *Cognitive modeling* (pp. 577–620). Cambridge, MA: MIT Press.]

Brown, J. S., & VanLehn, K. (1980). Repair theory: A generative theory of bugs in procedural skills. *Cognitive Science, 4,* 379–426.

Carbonell, J. G. (1983). Learning by analogy: Formulating and generalizing plans from past experience. In R. S. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach* (pp. 137–161). Palo Alto, CA: Tioga.

Carbonell, J. G. (1986). Derivational analogy: A theory of reconstructive problem solving and expertise acquisition. In R. S. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach* (Vol. 2, pp. 371–392). Los Altos, CA: Morgan Kauffmann.

Carroll, J. B. (1993). *Human cognitive abilities.* Cambridge, UK: Cambridge University Press.

Collins, A. M., & Loftus, E. F. (1975). A spreading-activation theory of semantic processing. *Psychological Review, 82,* 407–428.

Conway, F., & Siegelman, J. (2005). *Dark hero of the information age: In search of Norbert Wiener the father of cybernetics.* New York: Basic Books.

Corrigan-Halpern, A., & Ohlsson, S. (2002). Feedback effects in the acquisition of a hierarchical skill. In W. D. Gray & C.D. Schunn (Eds.), *Proceedings of the Twenty-Fourth Annual Conference of the Cognitive Science Society* (pp. 226–231). Mahwah, NJ: Lawrence Erlbaum.

Crevier, D. (1993). *AI: The tumultuous history of the search for artificial intelligence.* New York: Basic Books.

Crossman, E. (1959). A theory of the acquisition of speed-skill. *Ergonomics, 2,* 152–166.

Davis, R., & King, J. (1977). An overview of production systems. In E. Elcock & D. Michie (Eds.), *Machine intelligence* (Vol. 8, pp. 300–332). Chichester, UK: Horwood.

Doane, S. M., & Sohn, Y. W. (2000). ADAPT: A predictive cognitive model of user visual attention and action planning. *User Modeling and User-Adapted Interaction, 10,* 1–45.

Doane, S. M., Sohn, Y. W., McNamara, D. S., & Adams, D. (2000). Comprehension-based skill acquisition. *Cognitive Science, 24,* 1–52.

De Jong, K. (1990). Genetic-algorithm-based learning. In Y. Kodratoff & R. S. Michalski (Eds.), *Machine learning: An artificial intelligence approach* (Vol. 3, pp. 611–638). San Mateo, CA: Morgan Kauffmann.

Donald, M. (1991). *Origins of the modern mind: Three stages in the evolution of culture and cognition*. Cambridge, MA: Harvard University Press.

Ebbinghaus, H. (1885/1964). *Memory: A contribution to experimental psychology*. New York: Dover.

Elio, R., & Scharf, P. B. (1990). Modeling novice-to-expert shifts in problem-solving strategy and knowledge organization. *Cognitive Science, 14*, 579–639.

Engelberg, S. (2004). Lexical event structures for verb semantics. *Journal of Language and Linguistics, 3*, 62–108.

Ericsson, K. A., Charness, N., Feltovich, P. J., & Hoffman, R. R. (Eds.). (2006). *The Cambridge handbook of expertise and expert performance*. Cambridge, UK: Cambridge University Press.

Ericsson, K. A., Krampe, R. Th., & Tesch-Romer, C. (1993). The role of deliberate practice in the acquisition of expert performance. *Psychological Review, 100*, 363–406.

Falkenhainer, B., Forbus, K. D., & Gentner, D. (1989). The structure-mapping engine: Algorithm and examples. *Artificial Intelligence*, 41, 1–63.

Fauconnier, G., & Turner, M. (2002). *The way we think: Conceptual blending and the mind's hidden complexities*. New York: Basic Books.

Fikes, R. E., & Nilsson, N. J. (1993). STRIPS, a retrospective. *Artificial Intelligence, 59*, 227–232.

Fischer, K. W. (1980). A theory of cognitive development: The control and construction of hierarchies of skills. *Psychological Review, 87*, 477–531.

Fitts, P. (1964). Perceptual-motor skill learning. In A. Melton (Ed.), *Categories of human learning* (pp. 243–285). New York: Academic Press.

Forbus, K. D., Gentner, D., & Law, K. (1994). MAC/FAC: A model of similarity-based retrieval. *Cognitive Science*, 19, 141–205.

Fu, W.-T., & Gray, W. D. (2004). Resolving the paradox of the active user: Stable suboptimal performance in interactive tasks. *Cognitive Science*, 28, 901–935.

Gagne, R. M. (1970). *The conditions of learning* (2nd ed.). London, UK: Holt, Rinehart & Winston.

Gallistel, C. R. (1980). *The organization of action: A new synthesis*. Hillsdale, NJ: Lawrence Erlbaum.

Gärdenfors, P. (2000). *Conceptual spaces: The geometry of thought*. Cambridge, MA: MIT Press.

Gardner, H. (1985). *The mind's new science: A history of the cognitive revolution*. New York: Basic Books.

Gelman, R., & Gallistel, C. R. (1978). *The child's understanding of number*. Cambridge, MA: Harvard University Press.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science, 7*, 155–170.

Gernsbacher, M. A. (Ed.). (1994). *Handbook of psycholinguistics*. San Diego, CA: Academic Press.

Giunchiglia, E., Lee, J., Lifschitz, V., McCain, N. & Tuner, H. (2004). Nonmonotonic causal theories. *Artificial Intelligence, 153*, 49–104.

Greeno, J. G., Riley, M. S., & Gelman, R. (1984). Conceptual competence for children's counting. *Cognitive Psychology, 16*, 94–143.

Gray, W. D., & Boehm-Davis, D. A. (2000). Milliseconds matter: An introduction to microstrategies and to their use in describing and predicting interactive behavior. *Journal of Experimental Psychology: Applied, 6*, 322–335.

Gray, W. D., Schoelles, M. J., & Sims, C. R. (2005). Adapting to the task environment: Explorations in expected value. *Cognitive Systems Research, 6*, 27–40.

Hayes-Roth, F., Klahr, P., & Mostow, D. (1981). Advice taking and knowledge refinement: An iterative view of skill acquisition. In J. Anderson (Ed.), *Cognitive skills and their acquisition* (pp. 231–253). Hillsdale, NJ: Lawrence Erlbaum.

Hiebert, J., (Ed.). (1986). *Conceptual and procedural knowledge: The case of mathematics*. Hillsdale, NJ: Lawrence Erlbaum.

Hilgard, E. R., & Bower, G. H. (1966). *Theories of learning* (3rd ed.). New York: Appleton-Century-Crofts.

Holland, J. H. (1975). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. Ann Arbor: University of Michigan Press.

Holyoak, K. J. (1985). The pragmatics of analogical transfer. In G. H. Bower (Ed.), *The psychology of learning and motivation* (Vol. 19, pp. 59–87). New York: Academic Press.

Holyoak, K. J., & Thagard, P. R. (1989a). A computational model of analogical problem solving. In S. Vosniadou & A. Ortony (Eds.), *Similarity and analogical reasoning* (pp. 242–266). Cambridge, UK: Cambridge University Press.

Holyoak, K. J., & Thagard, P. (1989b). Analogical mapping by constraint satisfaction. *Cognitive Science, 13*, 295–355.

Huffman, S. B., & Laird, J. E. (1995). Flexibly instructable agents. *Journal of Artificial Intelligence Research, 3*, 271–324.

Hummel, J. E., & Holyoak, K. J. (1997). Distributed representations of structure: A theory of analogical access and mapping. *Psychological Review, 104*, 427–466.

Hummel, J. E., & Holyoak, K. J. (2003). A symbolic-connectionist theory of relational inference and generalization. *Psychological Review, 110*, 220–264.

Jackendoff, R. (1983). *Semantics and cognition.* Cambridge, MA: MIT Press.

Jain, N. K., & Bharadwaj, K. K. (1998). Some learning techniques in hierarchical censored production rules (HCPRs) system. *International Journal of Intelligent Systems, 13*, 319–344.

James, W. (1890). *Principles of psychology* (Vols. 1 & 2). London, UK: Macmillan.

Jones, G., Ritter, F. E., & Wood, D. J. (2000). Using a cognitive architecture to examine what develops. *Psychological Science, 11*, 93–100.

Jones, R. M., & Langley, P. A. (2005). A constrained architecture for learning and problem solving. *Computational Intelligence, 21*, 480–502.

Jones, R. M., & VanLehn, K. (1994). Acquisition of children's addition strategies: A model of impasse-free, knowledge-level learning. *Machine Learning, 16*, 11–36. [Reprinted in T. A. Polk & C. M. Seifert (Eds.). (2002). *Cognitive modeling* (pp. 623–646). Cambridge, MA: MIT Press.]

Keane, M. T., & Ledgeway, T., & Duff, S. (1994). Constraints on analogical mapping: A comparison of three models. *Cognitive Science, 18*, 387–338.

Kieras, D., & Bovair, S. (1986). The acquisition of procedures from text: A production-system analysis of transfer of training. *Journal of Memory and Language, 25*, 507–524.

Kintsch, W. (1998). *Comprehension: A paradigm for cognition.* Cambridge, UK: Cambridge University Press.

Kintsch, W., & Buschke, H. (1969). Homophones and synonyms in short-term memory. *Journal of Experimental Psychology, 80*, 403–407.

Koedinger, K. R., & Anderson, J. R. (1990). Abstract planning and perceptual chunks: Elements of expertise in geometry. *Cognitive Science, 14*, 511–550.

Kokinov, B. N., & Petrov, A. A. (2001). Integrating memory and reasoning in analogy-making: The AMBR model. In D. Gentner, K. J. Holyoak & B. N. Kokinov (Eds.), *The analogical mind: Perspectives from cognitive science* (pp. 59–124). Cambridge, MA: MIT Press.

Lakoff, G. (1971). On generative semantics. In D. D. Steinberg & L. A. Jakobovits (Eds.), *Semantics: An interdisciplinary reader in philosophy, linguistics and psychology* (pp. 232–296). Cambridge, UK: Cambridge University Press.

Langley, P. (1983). Learning search strategies through discrimination. *International Journal of Man-Machine Studies, 18*, 513–541.

Langley, P. (1985). Learning to search: From weak methods to domain-specific heuristics. *Cognitive Science, 9*, 217–260.

Langley, P. (1987). A general theory of discrimination learning. In D. Klahr, P. Langley, & R. Neches (Eds.), *Production system models of learning and development* (pp. 99–161). Cambridge, MA: MIT Press.

Langley, P., & Choi, D. (2006). Learning recursive control programs from problem solving. *Journal of Machine Learning Research, 7*, 493–518.

Larkin, J. H. (1981). Enriching formal knowledge: A model for learning to solve textbook physics problems. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition* (pp. 311–334). Hillsdale, NJ: Lawrence Erlbaum.

Lenat, D. B. (1983). Toward a theory of heuristics. In R. Groner, M. Groner & W. F. Bischof (Eds.), *Methods of heuristics* (pp. 351–404). Hillsdale, NJ: Lawrence Erlbaum.

Lewis, C. (1987). Composition of productions. In D. Klahr, P. Langley & R. Neches (Eds.), *Production system models of learning and development* (pp. 329–358). Cambridge, MA: MIT Press.

Lewis, C. (1988). Why and how to learn why: Analysis-based generalization of procedures. *Cognitive Science, 12*, 211–356.

Lifschitz, V., (Ed.). (1990). *Formalizing common sense: Papers by John McCarthy.* Norwoord, NJ: Ablex.

Logan, G. D. (1998). Toward an instance theory of automatization. *Psychological Review, 95*, 492–527.

Luchins, A. S., & Luchins, E. H. (1959). *Rigidity of behavior.* Eugene: University of Oregon Press.

Markman, A. B. (1999). *Knowledge representation*. Mahwah, NJ: Lawrence Erlbaum.

Martin, S. J., Grimwood, P. D., & Morris, R. G. M. (2000). Synaptic plasticity and memory: An evaluation of the hypothesis. *Annual Review of Neuroscience, 23*, 649–711.

Martin, S. J., & Morris, R. G. M. (2002). New life in an old idea: The synaptic plasticity and memory hypothesis revisited. *Hippocampus, 12*, 609–636.

McCarthy, J. (1959). Programs with common sense. In *Proceedings of the Teddington Conference on the Mechanization of Thought Processes* (pp. 75–91). London: Her Majesty's Stationary Office. [Reprinted as Section 7.1 of J. McCarthy, Programs with common sense, in Minsky (1968).]

McCarthy, J. (1963, July). *Situations, actions and causal laws*. Stanford Artificial Intelligence Project Memo No. 2. Stanford, CA: Stanford University. [Reprinted as Section 7.2 of J. McCarthy, Programs with common sense, in Minsky (1968).]

Miller, G. A., Galanter, E., & Pribram, K. H. (1960). *Plans and the structure of behavior*. New York: Holt, Rinehart & Winston.

Miller, G. A., & Johnson-Laird, P. N. (1976). *Language and perception*. Cambridge, MA: Harvard University Press.

Minsky, M., (Ed.). (1968). *Semantic information processing*. Cambridge, MA: MIT Press.

Mostow, D. J. (1983). Machine transformation of advice into a heuristic search procedure. In R. S. Michalski, J. G. Carbonell & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach* (pp. 367–404). Palo Alto, CA: Tioga.

Nason, S., & Laird, J. E. (2005). Soar-RL: integrating reinforcement learning with Soar. *Cognitive Systems Research, 6*, 51–59.

Neches, R. (1987). Learning through incremental refinement of procedures. In D. Klahr, P. Langley, & R. Neches (Eds.), *Production system models of learning and development* (pp. 163–219). Cambridge, MA: MIT Press.

Neches, R., Langley, P., & Klahr, D. (1987). Learning, development, and production systems. In D. Klahr, P. Langley, & R. Neches (Eds.), *Production system models of learning and development* (pp. 1–53). Cambridge, MA: The MIT Press.

Neimark, E. D., & Estes, W. K., (Eds.). (1967). *Stimulus sampling theory*. San Francisco: Holden-Day.

Nerb, J., Ritter, F. E., & Krems. J. F. (1999). Knowledge level learning and the power law: A Soar model of skill acquisition in scheduling. *Kognitionswissenschaft, 8*, 20–29.

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: Lawrence Erlbaum.

Newell, A. (1972). A theoretical exploration of mechanisms for coding the stimulus. In A. W. Melton & E. Martin (Eds.), *Coding processes in human memory* (pp. 373–434). New York: Wiley.

Newell, A. (1973). Production systems: Models of control structures. In W. G. Chase (Ed.), *Visual information processing* (pp. 463–526). New York: Academic Press.

Newell, A. (1990). *Unified theories of cognition*. Cambridge, MA: Harvard University Press.

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

Newell, A., Shaw, J. C., & Simon, H. A. (1958). Elements of a theory of human problem solving. *Psychological Review, 65*, 151–166.

Newell, A., & Simon, H. (1972). *Human problem solving*. Englewood Cliffs, NJ: Prentice Hall.

Oaksford, M., & Chater, N., (Eds.). (1998). *Rational models of cognition*. Oxford, UK: Oxford University Press.

Ohlsson, S. (1987a). Transfer of training in procedural learning: A matter of conjectures and refutations? In L. Bolc (Ed.), *Computational models of learning* (pp. 55–88). Berlin, West Germany: Springer-Verlag.

Ohlsson, S. (1987b). Truth versus appropriateness: Relating declarative to procedural knowledge. In D. Klahr, P. Langley, & R. Neches (Eds.), *Production system models of learning and development* (pp. 287–327). Cambridge, MA: MIT Press.

Ohlsson, S. (1992). Artificial instruction: A method for relating learning theory to instructional design. In P. Winne & M. Jones (Eds.), *Foundations and frontiers in instructional computing systems* (pp. 55–83). New York: Springer-Verlag.

Ohlsson, S. (1993). The interaction between knowledge and practice in the acquisition of cognitive skills. In S. Chipman & A. L. Meyrowitz (Eds.), *Foundations of knowledge* *acquisition: Cognitive models of complex learning* (pp. 147–208). Boston: Kluwer.

Ohlsson, S. (1996). Learning from performance errors. *Psychological Review, 103*, 241–262.

Ohlsson, S. (2007). The effects of order: A constraint-based explanation. In F. E. Ritter, J. Nerb, E. Lehtinen, & T. O'Shea (Eds.), *In order to learn: How the sequence of topics in fluences learing* (pp. 151–165). New York: Oxford University Press.

Ohlsson, S., Ernst, A. M., & Rees, E. (1992). The cognitive complexity of doing and learning arithmetic. *Journal of Research in Mathematics Education, 23*, 441–467.

Ohlsson, S., & Jewett, J. J. (1997). Ideal adaptive agents and the learning curve. In J. Brzezinski, B. Krause & T. Maruszewski (Eds.), *Idealization VIII: Modelling in psychology* (pp. 139–176) Amsterdam: Rodopi.

Ohlsson, S., & Rees, E. (1991a). The function of conceptual understanding in the learning of arithmetic procedures. *Cognition and Instruction, 8*, 103–179.

Ohlsson, S., & Rees, E. (1991b). Adaptive search through constraint violation. *Journal of Experimental and Theoretical Artificial Intelligence, 3*, 33–42.

Petrov, A. A., & Anderson, J. R. (2005). The dynamics of scaling: A memory-based anchor model of category rating and absolute identification. *Psychological Review, 112*, 383–416.

Piaget, J. (1950). *The psychology of intelligence.* London: Routledge & Kegan Paul.

Pirolli, P. (1986). A cognitive model and computer tutor for programming recursion. *Human-Computer Interaction, 2*, 319–355.

Pirolli, P. (1991). Effects of examples and their explanations in a lesson on recursion: A production system analysis. *Cognition and Instruction, 8*, 207–259.

Pirolli, P., & Recker, M. (1994). Learning strategies and transfer in the domain of programming. *Cognition & Instruction, 12*, 235–275.

Pustejovsky, J. (2005). *Meaning in context.* Cambridge, MA: MIT Press.

Pylyshyn, Z. W. (1979). Validating computational models: A critique of Anderson's indeterminacy of representation claim. *Psychological Review, 86*, 383–394.

Reason, J. (1990). *Human error.* Cambridge, UK: Cambridge University Press.

Restle, R. (1955). A theory of discrimination learning. *Psychological Review, 62*, 11–19.

Reimann, P., Schult, T. J., & Wichmann, S. (1993). Understanding and using worked-out examples: A computational model. In G. Strube & K. Wender (Eds.), *The cognitive psychology of knowledge* (pp. 177–201). Amsterdam: North-Holland.

Rosenbloom, P. S., Laird, J. E., & Newell, A., (Eds.). (1993). *The Soar papers: Research on integrated intelligence* (Vols. 1 & 2). Cambridge, MA: MIT Press.

Rosenbloom, P., & Newell, A. (1986). The chunking of goal hierarchies: A generalized model of practice. In R. S. Michalski, J. G. Carbonell & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach* (Vol. 2, pp. 247–288). Los Altos, CA: Kaufmann.

Rosenbloom, P., & Newell, A. (1987). Learning by chunking: A production system model of practice. In D. Klahr, P. Langley & R. Neches (Eds.), *Production system models of learning and development* (pp. 221–286). Cambridge, MA: MIT Press.

Ruiz, D., & Newell, A. (1993). Tower-noticing triggers strategy-change in the Tower of Hanoi: A Soar model. In P. S. Rosenbloom, J. E. Laird, & A. Newell (Eds.), *The Soar papers: Research on integrated intelligence* (Vol. 2, pp. 934–941). Cambridge, MA: MIT Press.

Russell, S., & Norvig, P. (1995). *Artificial intelligence: A modern approach.* Los Altos, CA: Morgan Kaufman.

Rychener, M. D. (1983). The instructible production system: A retrospective approach. In R. S. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach* (pp. 429–459). Palo Alto, CA: Tioga.

Rychener, M. D., & Newell, A. (1978). An instructible production system: Basic design issues. In D. A. Waterman & F. Hayes-Roth (Eds.), *Pattern-directed inference systems* (pp. 135–153). New York: Academic Press.

Ryle, G. (1949/1968). *The concept of mind.* London: Penguin.

Salomon, G., & Perkins, D. N. (1989). Rocky roads to transfer: Rethinking mechanisms of a neglected phenomenon. *Educational Psychologist, 24*, 113–142.

Salthouse, T. A. (1996). The processing-speed theory of adult age differences in cognition. *Psychology Review, 103*, 403–428.

Salvucci, D. D., & Anderson, J. R. (2001). Integrating analogical mapping and general problem solving: the path-mapping theory. *Cognitive Science, 25*, 67–110.

Schooler, L. J., & Hertwig, R. (2005). How forgetting aids heuristic inference. *Psychological Review, 112*, 610–628.

Schneider, W., & Chein, J. M. (2003). Controlled & automatic processing: Behavior, theory, and biological mechanisms. *Cognitive Science, 27*, 525–559.

Schneider, W., & Oliver, W. L. (1991). An instructable connectionist/control architecture: Using rule-based instructions to accomplish connectionist learning in a human time scale. In K. VanLehn (Ed.), *Architectures for intelligence* (pp. 113–145). Hillsdale, NJ: Lawrence Erlbaum.

Shrager, J., Hogg, T., & Huberman, B. A. (1988). A graph-dynamic model of the power low of practice and the problem-solving fan effect. *Science, 242*, 414–416.

Shrager, J., & Siegler, R. S. (1998). A model of children's strategy choices and strategy discoveries. *Psychological Science, 9*, 405–410.

Siegler, R., & Araya, R. (2005). A computational model of conscious and unconscious strategy discovery. In R. V. Kail (Ed.), *Advances in child development and behavior* (Vol. 33, pp. 1–42). Oxford, UK: Elsevier.

Siegler, R. S., & Shipley, C. (1995). Variation, selection, and cognitive change. In T. J. Simon & G. S. Halford (Eds.), *Developing cognitive competencies: New approaches to process modeling* (pp. 31–76). Hillsdale, NJ: Lawrence Erlbaum.

Siegler, R. S., & Shrager, J. (1984). Strategy choices in addition and subtraction: How do children know what to do? In C. Sophian (Ed.), *Origins of cognitive skills* (pp. 229–293). Hillsdale, NJ: Lawrence Erlbaum.

Simon, H. A. (1956). Rational choice and the structure of the environment. *Psychological Review, 63*, 129–138.

Simon, H. A. (1970). *The sciences of the artificial.* Cambridge, MA: MIT Press.

Simon, H. A. (1972). On reasoning about actions. In H. A. Simon & L. Siklossy, (Eds.), *Representation and meaning* (pp. 414–430). Englewood Cliffs, NJ: Prentice Hall.

Singley, M. K., & Anderson, J. R. (1989). *The transfer of cognitive skill.* Cambridge, MA: Harvard University Press.

Smith, D. A., Greeno, J. G., & Vitolo, T. M. (1989). A model of competence for counting. *Cognitive Science, 13*, 183–211.

Sohn, Y. W., & Doane, S. M. (2002). Evaluating comprehension-based user models: Predicting individual user planning and action. *User Modeling and User-Adapted Interaction, 12*, 171–205.

Spellman, B. A., & Holyoak, K. J. (1996). Pragmatics in analogical mapping. *Cognitive Psychology, 31*, 307–346.

Sun R., Merrill, E., & Peterson, T. (2001). From implicit skills to explicit knowledge: A bottom-up model of skill learning. *Cognitive Science, 25*, 203–244.

Sun, R., Slusarz, P., & Terry, C. (2005). The interaction of the explicit and the implicit in skill learning: A dual-process approach. *Psychological Review, 112*, 159–192.

Taatgen, N. A. (2002). A model of the individual differences in skill acquisition in the Kanfer-Ackerman air traffic control task. *Cognitive Systems Research, 3*, 103–112.

Taatgen, N. A. (2005). Modeling parallelization and flexibility improvements in skill acquisition: From dual tasks to complex dynamic skills. *Cognitive Science, 29*, 421–455.

Taatgen, N. A., & Anderson, J. R. (2002). Why do children learn to say "Broke"? A model of learning the past tense without feedback. *Cognition, 86*, 123–155.

Taatgen, N. A., & Lee, F. J. (2003). Production compilation: A simple mechanism to model complex skill acquisition. *Human Factors, 45*, 61–76.

Thorndike, E. L. (1898). *Animal intelligence: An experimental study of the associative processes in animals.* Unpublished doctoral dissertation, Columbia University, New York.

Thorndike, E. L. (1911). *The principles of teaching based on psychology.* New York: A. G. Seiler.

Thorndike, E. L. (1927). The law of effect. *American Journal of Psychology, 39*, 212–222.

VanLehn, K. (1983, November). *Felicity conditions for human skill acquisition: Validating an AI based theory* (Technical Report CIS 21). Palo Alto, CA: Xerox Palo Alto Research Centers.

VanLehn, K. (1987). Learning one subprocedure per lesson. *Artificial Intelligence, 31*, 1–40.

VanLehn, K. (1988). Toward a theory of impasse-driven learning. In H. Mandl & A. Lesgold (Eds.), *Learning issues for intelligent tutoring systems* (pp. 19–41). New York: Springer-Verlag.

VanLehn, K. (1990). *Mind bugs: The origins of procedural misconceptions.* Cambridge, MA: MIT Press.

VanLehn, K. (1999). Rule-learning events in the acquisition of a complex skill: An evaluation of Cascade. *The Journal of the Learning Sciences, 8*, 71–125.

VanLehn, K., & Brown, J. S. (1980). Planning nets: A representation for formalizing analogies and semantic models of procedural skills. In R. E. Snow, P.-A. Federico & W. E. Montague (Eds.), *Aptitude, learning, and instruction: Cognitive process analyses of learning and problem solving* (Vol. 2, pp. 95–137). Hillsdale, NJ: Lawrence Erlbaum.

VanLehn, K., & Jones, R. (1993). Learning by explaining examples to oneself: A computational model. In S. Chipman & A. L. Meyrowitz (Eds.), *Foundations of knowledge acquisition: Cognitive models of complex learning* (pp. 25–82). Boston: Kluwer.

VanLehn, K., Jones, R. M., & Chi, M. T. H. (1992). A model of the self-explanation effect. *The Journal of the Learning Sciences, 2*, 1–59.

VanLehn, K., Ohlsson, S., & Nason, R. (1994). Applications of simulated students: An exploration. *Journal of Artificial Intelligence and Education, 5*, 135–175.

Veloso, M. M., & Carbonell, J. G. (1993). Derivational analogy in Prodigy: Automating case acquisition, storage and utilization. *Machine Learning, 10*, 249–278.

Waterman, D., & Hayes-Roth, F. (1978). An overview of pattern-directed inference systems. In D. Waterman & F. Hayes-Roth (Eds.), *Pattern-directed inference systems* (pp. 3–22). New York: Academic Press.

Watson, J. B. (1913). Psychology as the behaviorist views it. *Psychological Review, 20*, 158–177.

Welford, A. T. (1968). *Fundamentals of skill.* London: Methuen.

Wickelgren, W. A. (1969). Auditory or articulatory coding in verbal short-term memory. *Psychological Review, 76*, 232–235.

Wilson, W. H., Halford, G. S., Gray, B., & Phillips, S. (2001). The STAR-2 model for mapping hierarchically structured analogs. In D. Gentner, K. J. Holyoak & B. N. Kokinov (Eds.), *The analogical mind: Perspectives from cognitive science* (pp. 125–159). Cambridge, MA: MIT Press.

Winograd, T. (1975). Frame representations and the declarative/procedural controversy. In D. Bobrow & A. Collins (Eds.), *Representation and understanding: Studies in cognitive science* (pp. 185–210). New York: Academic Press.

Winston, P. H. (1986). Learning by augmenting rules and accumulating censors. In R. S. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach* (Vol. 3, pp. 45–61). Los Altos, CA: Kaufmann.

Wolpert, D. M., Ghahramani, Z., & Flanagan, J. R. (2001). Perspectives and problems in motor learning. *Trends in Cognitive Sciences, 5*, 487–494.

Woltz, D. J., Gardner, M. K., & Bell, B. G. (2000). Negative transfer errors in sequential skills: Strong-but-wrong sequence application. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 26*, 601–625.

Woodworth, R. S. (1938). *Experimental psychology.* New York: Henry Holt.