Synthetic Reasoning and the Reverse Engineering of Boolean Circuits

N. Y. Louis Lee (ngarlee@princeton.edu)
Department of Psychology, Princeton University
Princeton, NJ 08544-1010 USA

P. N. Johnson-Laird (phil@princeton.edu)
Department of Psychology, Princeton University
Princeton, NJ 08544-1010 USA

Abstract

In synthetic reasoning, individuals assemble elementary components into effective systems, such as the working mechanism of an unknown device. This paper proposes a new theory of this ability, and reports two experiments investigating how individuals reverse engineer Boolean circuits with two inputs and an output. Experiment 1 supported the theory's prediction that the complexity, and hence difficulty, of synthetic reasoning problems should depend on the number of possibilities in which the assembled system works, the number of components in that system, and the relations between the component parts. Experiment 2 generalized this finding, and showed that individuals develop two distinct strategies.

Introduction

Synthetic reasoning is a sequence of mental steps that individuals follow in assembling elementary components into an effective system. When you explain an everyday event, you synthesize your existing causal knowledge with new information in order to explain the event. When you figure out how a device works, you infer from the functions of each of the device's components the overall mechanism. Synthetic reasoning calls for both deduction and induction, especially the form of induction that generates explanations, i.e., "abduction". It occurs both in daily life and science. But, how do people do it?

Cognitive scientists have investigated a variety of aspects of synthetic reasoning in both psychology and artificial intelligence (e.g., Johnson & Krems, 2001). Klahr and colleagues have studied how individuals discover the function of a control on a toy robot (see, e.g., Klahr & Dunbar, 1988; Klahr, 2000). The participants write programs that control the robot, to try to discover the function of the control. The main finding was that individuals differ in whether they focus on hypotheses about the control or on possible experiments. AI researchers have proposed accounts of 'abductive' reasoning in which individuals generate explanations (for a review, see Paul, 1993). These accounts, however, presuppose a pre-existing set of putative explanations, i.e., they have finessed the problem of how individuals use knowledge to synthesize explanations. For example, the 'set-cover' approach selects subsets of existing hypotheses, e.g., Allemang, Tanner, Bylander, & Josephson (1987). Similarly, the 'explanatory-coherence' account relies on a handcrafted connectionist model that represents competing hypotheses, e.g., Thagard (2000).

Hence, despite a sizable literature in explanatory reasoning and abduction, the underlying mental processes of synthetic reasoning remain largely unknown. We therefore formulated a theory of synthetic reasoning, and carried out two experiments to test it. The next section describes our theory and illustrates our test-bed of Boolean systems. A Boolean system, such as an electrical circuit of switches, has a "logic" equivalent to negation, conjunction, and disjunction. This logic also applies to concepts (e.g., Shepard, Hovland, & Jenkins, 1961), to sentential connectives (e.g., Johnson-Laird, Byrne, & Schaeken, 1992), and to learning algorithms in artificial intelligence (e.g., Kearns & Vazirani, 1994). No-one knows for certain what makes Boolean problems difficult. Our theory, however, makes clear predictions about their difficulty.

A Theory of Synthetic Reasoning

In order to construct a working model of a system, you need to understand what the system does and how its components work. Our theory postulates that individuals construct *mental models* of systems, i.e., representations in which the structure of the model corresponds to the structure of the system (Gentner & Stevens, 1983; Johnson-Laird, 2001). But, how do individuals construct such a model? Like any sort of thinking – with the possible exception of mental arithmetic – the process of synthetic reasoning has to be treated as non-deterministic (Hopcroft & Ullman, 1979). As in deductive reasoning (van der Henst, Yang, & Johnson-Laird, 2002) and problem solving (Lee & Johnson-Laird, 2004), reasoners should develop different strategies as they learn to synthesize systems of the same sort. There are two main sorts of strategies that they are likely to develop: they may focus one at a time on the possibilities in which the system either does or does not produce an output, or they may focus on each of the input components one at a time and try to account for its effects on the output. To grasp the difference between the two strategies, consider the following problem in which individuals have to assemble an electrical circuit containing two binary switches, a battery, a light bulb, and some wires. In this circuit, the light comes on when one or both of the switches are up. Thus, the circuit has four different possibilities:

Switch A	Switch B	Light
Up	Up	On
Up	Down	On
Down	Up	On
Down	Down	Off

In the first sort of strategy, individuals try to account for each possible outcome one at a time. In deductive reasoning, individuals typically focus on what is true, but not what is false (see, e.g., Johnson-Laird & Savary, 1999). Hence, they should be more likely to consider first the positive possibilities in which the light comes on, rather than the possibilities in which it does not come on. They accordingly construct a circuit that accounts for the first positive possibility (e.g., when both switches are up, and the light comes on), and then modify the circuit to account for the remaining possibilities. In the second strategy, they consider the effects on the light of each switch separately. For example, they notice that when one switch is up, the light always comes on, and so they construct a circuit with only one switch that connects the ciruit. Then, they work out how to insert another switch into the circuit so as to account for all the possibilities. Both strategies ultimately require individuals to make sure that all the components yield the correct outcomes.

This theory of strategies predicts that three factors should affect the difficulty of synthetic problems. The first factor is the number of variable components that the system contains. A variable component is a component that has more than one state, e.g., a switch. This factor is similar to Halford's concept of *relational complexity*, which he regards as sufficient to account for complexity (see, e.g., Halford, Wilson, & Philips, 1998). The prediction, say, that a system with two components should be easier to synthesize than one with two hundred components hardly warrants an experimental test.

A second factor is the number of positive possibilities, i.e., the possibilities in which the light comes on. Many studies of reasoning have demonstrated such effects (see, e.g., Johnson-Laird, 2001). Hence, we can predict that the *or* problem in the table above should be harder to synthesize than an *and* problem with only one positive possibility.

The third factor is more subtle. It is the dependence of the input components on one another in yielding the output. In the case of the *or* problem above, each switch acts independently of the other to switch the light on. In the case of an *and* problem (see below), each switch acts independently of the other to switch the light off. In contrast, an *or-else* problem (see below) is a dependent one. In this problem, the light comes on only when one *or-else* the other, but not both of the switches, is up. Hence, the effect of one switch depends on the other switch's position. This notion of dependence is similar to Vapnik's (1998) notion of a non-linear system. But, unlike linearity versus non-linearity, dependence is a gradeable notion. Imagine a system controlled by three switches. If one particular switch makes the light come on in, say, three out of the four positive possibilities, the switch is relatively independent of the others. Hence, the problem should be easier than one in which none of the switches has this privileged effect.

Granted that individuals tend to focus on the system's possibilities or on its input components, independent systems should be easier to reverse engineer than dependent ones. In sum, three factors should determine the difficulty of synthesizing a system, at least a Boolean system: the number of variable input components, the number of positive possibilities, and the relative independence of the input components. To test the theory, we carried out two experiments calling for the reverse engineering of Boolean systems.

Experiment 1

Experiment 1 examined the reverse engineering of three sorts of Boolean electrical circuit. On each trial, the participants saw a "black box" with two switches and a bulb. The computer displayed the four possible switch settings and whether or not the bulb came on. Figure 1 presents such a problem. The participants' task was to design the circuit connecting the switches that yielded these contingencies. In the *and* problem, the bulb came on only when both switches were up (as in Figure 1). In the *or* problem, the bulb came on when one or other switch was up or both of them were. In the *or-else* problem, the bulb came on only when one or other of the switches was up, but not both. Figure 2 shows the minimal solutions of the three problems in the experiment.

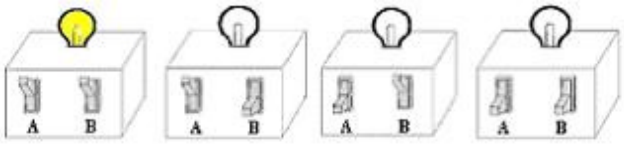

Figure 1: The presentation of the *and* problem in Experiment 1. This picture shows the four different combinations of the switch positions and their effects on the bulb. It comes on only when both switches are up.

The theory predicts that the two independent problems (*and* and *or*) should be easier than the dependent problem (*or-else*), because dependence plays havoc with the two strategies that we described earlier. You cannot focus on one input component at a time. The theory also predicts that the *and* problem (one positive possibility) should be easier than the *or* problem (three positive possibilities). The ease of each problem should be reflected in the accuracy of the circuits, fewer separate drawings to produce a correct solution, and a faster time to produce it.

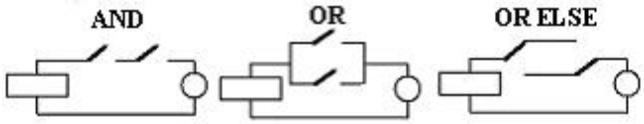

Figure 2: The minimal Boolean electrical circuits for *and*, *or*, and *or-else* problems. The circles represent bulbs, the rectangles represent batteries, and the switches are binary.

Method and Procedure

We tested 18 Princeton University students individually. The experimenter explained that their task was to design a circuit for a "black box" that contained the following components: a battery, a bulb, two binary switches, and as many wires as necessary. Each switch had one input terminal and either one or two output terminals, so that the switch could make or break one or two circuits. The experimenter explained: a switch can function in two ways. A "simple" switch uses one output terminal and closes or breaks a single circuit; whereas a "complex" switch has two output terminals so that in one position it closes one circuit whilst breaking the other circuit, and in the other position it has the opposite effect. The experimenter illustrated with examples how both sorts of switches worked. The aim of a circuit was to produce the effects of the switches' positions on the light. The participants carried out a practice trial for a black box with one switch and one bulb. The experimenter answered the participants' questions, and then proceeded to the experiment proper. The participants drew diagrams of the circuits and they were encouraged to draw as many as they needed on the answer sheet. The experimenter told them that they had seven minutes to solve each problem, and that they would be timed. The instructions and the problems were presented using the PowerPoint program, and each participant received the three problems in one of the six possible orders.

Results and Discussion

Figure 3 presents the number of correct responses and the overall mean latencies. The figure shows the predicted trend: the *and* problem yielded more correct solutions than the *or* problem, which in turn yielded more correct solutions than the *or-else* problem (Page's L = 237.0, z = 3.50, p <.01). Likewise, the predicted trend occurred in the times to solve the problems (Page's L = 234.5, z = 3.08, p <.005). The mean numbers of diagrams that the participants drew to reach a solution or to exceed the time limit were 1.1, 2.4, and 3.8 diagrams for the *and*, *or*, and *or-else* problems respectively, and this trend was also reliable (Page's L = 194.0, z = 3.67, p <.0005). The results accordingly corroborated the theory.

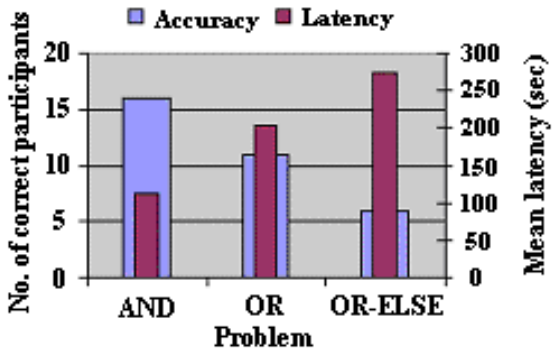

Figure 3: Mean latencies (of both accurate and inaccurate responses) and numbers of accurate responses in the three problems of Experiment 1.

Experiment 2

The aim of Experiment 2 was to examine the strategies that individuals developed as they reverse engineered Boolean problems. It also aimed to generalize the results to a new domain of water flow systems. The task in this domain was to assemble a water flow system from the following components: a pump that supplied the water, two faucets, a turbine, and pipes that were either straight or L-shaped. Figure 4 shows the presentation of the *and* problem. The task was to design a system that ensured that the turbine ran only with the appropriate positions of the faucets. The three problems in this domain were isomorphic to those in Experiment 1; Figure 5 presents correct minimal solutions. The theory predicts that the participants should employ the two principal strategies that we described earlier, either focusing on one input component (i.e., binary switch) at a time, or one outcome possibility at a time. It also predicts the same trend of difficulty for both the electrical and water flow problems. Even though the participants receive no feedback, the second block of problems should be easier than the first block.

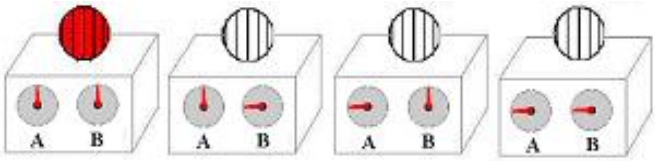

Figure 4: The presentation of the *and* problem in a water flow system in Experiment 2. This picture shows the four different combinations of the faucets' positions and their effects on the turbine. It comes on, as shown in red, only when both faucets are up.

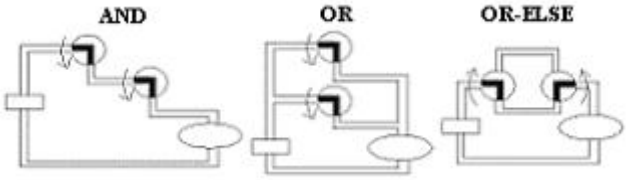

Figure 5: The minimal Boolean water flow solutions for the *and*, *or*, and *or-else* problems. The ellipses represent turbines, the rectangles represent water pumps, and the faucets are binary.

Method and Procedure

We tested 20 Princeton University students with two blocks of three problems. One block contained the three electrical circuit problems; and the other block contained the three water flow problems. The order of the two blocks, and the order of the problems within each block, were counter-balanced over the participants. The procedure was the same as in Experiment 1, with a training trial before each block of problems. However, in this experiment, the participants had to think aloud as they solved the problems, they had 11 minutes to solve each problem, and they had to describe the strategies that they had used in a post-experiment interview.

We recorded their protocols with a portable cassette tape recorder.

Results and Discussion

The numbers of correct solutions corroborated the predicted trend in both domains: the *and* problem yielded more correct solutions (20 in both domains) than the *or* problem (9 in both domains), which in turn yielded more correct solutions than the *or-else* problem (5 in both domains, Page's L = 269.0, z = 4.59, p <<.001; it was also highly reliable for each domain separately). Figure 6 presents the mean times that the participants spent on the problems, whether the solution was correct or incorrect. These trends were highly reliable (Page's L = 272.5, z = 5.14, p <<.001). The trend was also reliable in the number of diagrams that the participants drew (means, excluding the final diagram, were 0.88, 4.28, and 4.88 for the *and*, *or*, and *or-else* problems respectively, Page's L = 270.0, z = 4.74, p<<.001). No reliable differences occurred in either accuracies or latencies between the two domains, or even between the two blocks of trials.

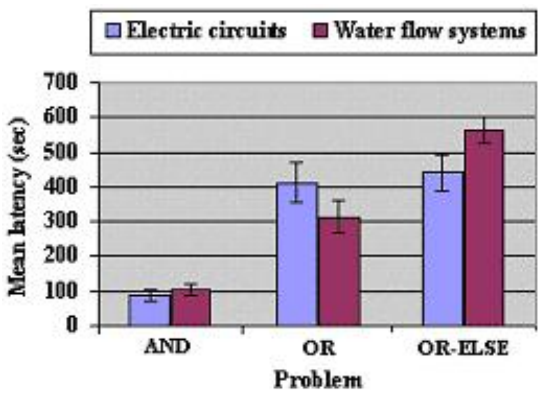

Figure 6: Mean latencies (of both accurate and inaccurate responses) for the three problems in each domain in Experiment 2.

An analysis of the participants' protocols showed that they indeed developed two main strategies for synthesizing the systems. These strategies were also borne out by their post-experimental interviews. They did not necessarily use the same strategy for all problems, and some participants even switched from one strategy to another whilst they were solving a problem.

As predicted, the first strategy was to consider each input-output possibility separately. The participants first synthesized a solution for one possibility, and then tried to modify this solution to capture the other possibilities. Figure 7 illustrates how one participant (No.3) used this strategy to solve the water flow *or* problem. She started with the possibility in which both switches were up and the turbine was running, and constructed a working model for this possibility (1). She modified this model to capture the possibility in which faucet A was up and faucet B was not, and the turbine was running. She accordingly added a branching pipe between the two faucets, which she connected to the other end of faucet B so that the system would still be closed when switch B was not up (2). She repeated this step to account for the third possibility in which faucet A was not up and faucet B was up, and the turbine was running. This solution, however, was incorrect, because the turbine would still come on when both switches were not up.

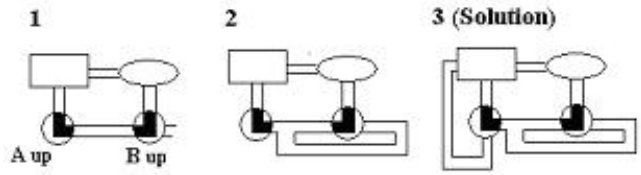

Figure 7: An example of the strategy of building a model to account for one possibility first, and then modifying to account for the other possibilities (see text).

The second strategy was to focus on the effects of a single switch or faucet. Figure 8 shows how a participant (No.16) used this strategy to solve the electrical *or* problem. She first focused on the fact that the bulb always came on when switch A was up. She drew a complete circuit with only switch A (2). She then considered the possibilities in which switch A was down (3), and added switch B to the model (4). Then, she worked out which output terminals in the circuit corresponded with which switch positions, changing her mind about what the "up" position of switch A (5-6) and the "up" position of switch B (6-8) should be. Finally, she drew out the resulting model in full as a correct solution (9).

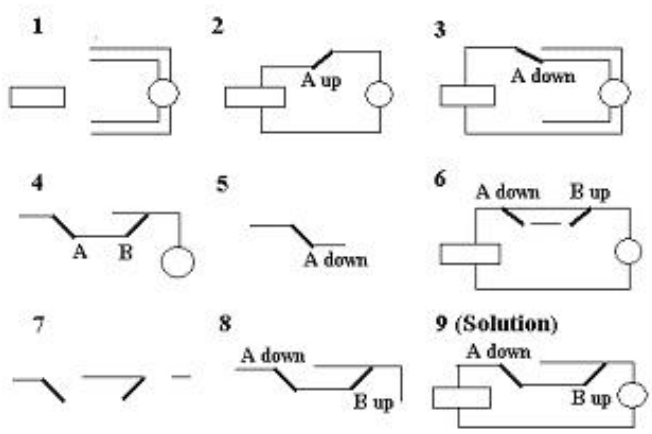

Figure 8: An example of the second strategy in which the participant focuses on the effects of a single switch on the bulb, and then extends the model to account for the second switch (see text).

General Discussion

When individuals try to reverse engineer a system, they need to understand what the system does and how its components work. They then have to assemble the components in a synthesis that delivers the required performance of the system. As the theory predicts, individuals tend to synthesize a circuit by accounting either for one switch or else one possibility at a time. The difficulty of the task follows from

this account: it depends on the number of variable components, the number of settings of them that yield positive outputs from the system, and the dependence of the system's input components. Experiment 1 showed that both the number of positive outputs and dependence affected performance. The participants found it easier to synthesize an *and* circuit (one positive possibility) than an *or* circuit (three positive possibilities). Even though an *or-else* circuit has only two positive possibilities, it was hardest of all, because the two input components' effects are dependent. Their positions interact. The difficulty in synthesizing an exclusive disjunction is well known to "connectionists": unlike inclusive disjunction, it calls for "hidden units" if a network of units is to learn it (see McClelland & Rumelhart, 1986).

Experiment 2 corroborated these findings, showing that the same trend occurred in synthesizing water flow systems. More importantly, the participants' think-aloud protocols and the post-experiment interviews showed that they did develop two principal strategies. In one, the participants focus on a single positive possibility, and construct a circuit for it. They then try to extend the circuit to cope with the other possibilities. In the other, they focus on a single switch, and construct a circuit that controls the output appropriately. They then try to extend the circuit to cope with the effects of the other switch.

Are alternative theories likely to account for our results? The most influential psychological theory views the search for a solution to a problem as governed by a means-ends analysis (see, e.g., Newell & Simon, 1972; Newell, 1990). But, as in many other synthetic tasks, our circuit problems are not amenable to this heuristic. The goal of our problems was clear, but it was not one that allowed our participants to envisage a position that was just a move away from the solution. Hence, they cannot work backwards from the goal. Likewise, because both the number of positive possibilities and the dependence of the input components affected the difficulty of the problems, it seems unlikely that relational complexity alone can account for performance (*pace* Halford et al., 1998). In addition, some theories argue that the complexity of Boolean concepts depends on the concept's minimal description, i.e., the length of the concept's shortest equivalent logical formula (see e.g., Feldman, 2000). However, such accounts treat both conjunction and inclusive disjunction, but not exclusive disjunction, as the primitives of a logical formula. Hence, although these accounts can predict why an *or-else* problem should be harder than an *or* or an *and* problem, they do so merely by stipulating that or-else is not allowed in their descriptive language. They also fail to explain why an *or* problem should be harder than an *and* problem.

Readers may argue that the *or-else* problems call for an *insight* (see, e.g., Knoblich, Ohlsson, Haider, & Rhenius, 1999; Ormerod, MacGregor, & Chronicle, 2002), namely, the realization that instead of breaking or closing a single circuit, a switch can also direct the current into two different circuits in its two different positions. In our view, this possibility does not explain the difficulty of the *or-else* problems, if only because the participants were explicitly taught this use of the switches, and because some participants used this switch in solving the inclusive *or* problems. But, we cannot as yet rule out this putative explanation. The model-based theory of synthetic reasoning has so far yielded reliable predictions. The theory extends to domains outside Boolean circuits, but we have yet to test its applications there.

Acknowledgments

This research was supported by a grant from the National Science Foundation to the second author to study strategies in reasoning (BCS-0076287). We thank Caren Frosch, Sonja Geiger, Sam Glucksberg, Geoff Goodwin, Cathy Haught, and Ira Noveck for helpful comments.

References

Allemang, D., Tanner, M., Bylander, T., & Josephson, J. (1987). Computational complexity of hypothesis assembly. *Proceedings of the 10th International Joint conference on Artificial Intelligence*, pp.1112-1117.

Feldman, J. (2000). Minimization of Boolean complexity in human concept learning. *Nature, 407*, 630-634.

Gentner, D., & Stevens, A.L. (Eds.) (1983). *Mental models*. NJ: LEA.

Halford, G.S., Wilson, W.H., & Phillips, S. (1998). Processing capacity defined by relational complexity: Implications for comparative, developmental, and cognitive psychology. *Behavioral and Brain Sciences, 21*, 803-831.

Hopcroft, J.E., & Ullman, J.D. (1979). *Formal languages and their relation to automata*. Reading, MA: Addison-Wesley.

Johnson, T.R., & Krems, J.F. (2001). Use of current explanations in multicausal abductive reasoning. *Cognitive Science, 25*, 903-939.

Johnson-Laird, P.N. (2001). Mental models and deduction. *Trends in Cognitive Sciences, 5*, 434-442.

Johnson-Laird, P.N., Byrne, R.M.J., & Schaeken, W. (1992). Propositional reasoning by model. *Psychological Review, 99,* 418-439.

Johnson-Laird, P.N., & Savary, F. (1999). Illusory inferences: A novel class of erroneous deductions. C*ognition*, *71*, 191-229.

Kearns, M.J., & Vazirani, U.V. (1994) *An Introduction to Computational Learning Theory*. Cambridge, MA: MIT Press.

Klahr, D. (2000). *Exploring science: The cognition and development of discovery processes.* Cambridge, MA: MIT Press.

Klahr, D., & Dunbar, K. (1988). Dual space search during scientific reasoning. *Cognitive Science, 12*, 1-48.

Knoblich, G., Ohlsson, S., Haider, H., & Rhenius, D. (1999). Constraint relaxation and chunk decomposition in insight problem solving. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 25*, 1534-1555.

Lee, N.Y.L., & Johnson-Laird, P.N. (2004). Creative strategies in problem solving. In K. Forbus, D. Gentner & T. Regier (Eds.). *Proceedings of the Twenty-Sixth Annual*

Conference of the Cognitive Science Society, Chicago, IL (pp.813-818). Mahwah, NJ: LEA.

Rumelhart, D. E., & McClelland, J. L. (1986). *Parallel distributed processing: Explorations in the microstructure of cognition, (Vol. 1 & 2)*. Cambridge, MA: MIT Press.

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

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

Ormerod, T.C., MacGregor, J.N., & Chronicle, E.P. (2002). Dynamics and constraints in insight problem solving. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 28, 791-799.

Paul, G. (1993). Approaches to abductive reasoning: an overview. *Artificial Intelligence Review, 7*, 109-152.

Shepard, R. N., Hovland, C. I., & Jenkins, H. M. (1961). Learning and memorization of classifications. *Psychological Monographs: General and Applied*, *75*, 1-42.

Thagard, P. (2000). *Coherence in Thought and Action*. Cambridge, MA: MIT Press.

Van der Henst, J.-B., Yang, Y., & Johnson-Laird, P.N. (2002). Strategies in sentential reasoning. *Cognitive Science, 26*, 425-468.

Vapnik, V.N. (1998). Statistical learning theory. New York: John Wiley & Sons.