CHAPTER 12

Mental Logic, Mental Models, and Simulations of Human Deductive Reasoning

Philip N. Johnson-Laird and Yingrui Yang

1. Introduction

Individuals who know no logic are able to make deductive inferences. Given a problem such as:

> If the printer test was run then the printer produced a document.
> The printer test was run.
> What follows?

they draw the conclusion:

> The printer produced a document.

The conclusion is the result of a *valid* deduction, that is, if the premises are true, then the conclusion must be true also. How naive individuals – those untrained in logic – are able to draw valid conclusions is a matter of controversy, because no one has access to the mental processes underlying inferences. Some cognitive scientists believe that these processes are analogous to those of "proof" theory in logic (see Chapter 5 in this volume on logic-based modeling); some believe that they are analogous to those of "model" theory in logic; and some believe that logic is irrelevant and that the probability calculus is a better guide to human deductive reasoning. The present chapter focuses on simulations based on proof theory and model theory, but it has something to say about the probabilistic theory.

The chapter starts with an outline of how psychological theories based on formal rules of inference – proof theory, that is – can be implemented to simulate reasoning. It uses as a test-bed so-called sentential reasoning based on negation and connectives, such as "if," "and," and "or." This sort of reasoning lies at the heart of our everyday deductions, although we are soon defeated by complex inferences in this domain. The chapter then turns to programs simulating the theory inspired by "model" theory in logic, that is, the theory of *mental* models, which posits that the engine of human reasoning relies on content. It illustrates two simulations of the theory. One program simulates spatial reasoning, and it shows how valid inferences can be drawn without explicit representations of the logical properties of relations. Instead, they emerge from the representations of the

meanings of relational terms. The other program concerns sentential reasoning, and it shows how an apparently unexceptional assumption leads to a striking prediction of systematic fallacies in reasoning – a case that yields crucial predictions about the nature of human deductive reasoning. The chapter concludes with an attempt to weigh up the nature of human rationality in the light of these and other simulation programs.

2. The Simulation of Formal Theories of Reasoning

For many years, psychologists argued that deduction depends on an unconscious system of formal rules of inference akin to those in proof-theoretic logic. The inference in the opening example, for example, could be drawn using the formal rule of inference known as *modus ponens*:

If A then B
A
Therefore, B.

The human inference engine matches the form of the premises to this rule, where *A* has as its value: *the printer test was run*, and *B* has as its value: *the printer produced a document*. The use of the rule proves the conclusion, *B*. This sort of theory has its proponents both in artificial intelligence and in cognitive psychology. Its intellectual godfather in psychology was the Swiss theorist, Jean Piaget (see, e.g., Beth & Piaget, 1966), but many theorists have proposed versions of the doctrine (e.g., Braine, 1978; Braine & O'Brien, 1998; Johnson-Laird, 1975; Osherson, 1974–1976; Rips, 1983, 1994).

Rips (1994) describes an implementation of his version of the theory, and the proponents of the other leading formal rule theory (Braine & O'Brien, 1998) have described an algorithm for it, although they did not implement a program. Hence, this section focuses on Rips's (1994) program. He argues that formal rules, such as modus ponens, are central to human cognition, underlying not just deduction but all thinking. Hence, formal rules on his account are part of cognitive architecture and akin to a general-purpose programming system in which any sort of theory can be implemented, even, say, Newell's (1990) Soar theory (see Chapter 6 in this volume on cognitive architecture). Soar is a so-called production system, which is made up of a large number of productions, that is, conditional rules with specific contents. They have the form: *if condition X holds then carry out action Y*, and a production can be triggered whenever its antecedent condition is satisfied. Rips argues that this method of applying the rules is akin to the use of modus ponens, but that Newell's theory is "too unconstrained to explain what is essential about deduction" (Rips, 1994, p. 30).

At the heart of Rips's (1994) theory is the notion of a mental proof, so theorists need to devise psychologically plausible formal rules of inference and a psychologically plausible mechanism to use them in constructing mental proofs. Like several proposals in the mid-1970s (e.g., Braine, 1978; Johnson-Laird, 1975; Osherson, 1974–1976), Rips adopts the "natural deduction" approach to rules of inference. Each logical connective has its own rules. Each quantifier, such as "every" and "some," also has its rules, too, although Rips presupposes an input to the program that captures the logical form of premises (see Chapter 5 in this volume on logic-based modeling). This section accordingly focuses on Rips's system for reasoning with sentential connectives. It has rules to introduce each connective into a proof, for example:

A
B
A and B

where the proposition beneath the line signifies the conclusion. And the system has rules to eliminate connectives, for example:

If A then B
A
B.

Natural deduction can yield intuitive proofs, and it was popular in logic texts, although the so-called "tree" method supplanted it (e.g., Jeffrey, 1981). Rips refers to the tree method, which simulates the search for counterexamples, but he considers it to be psychologically implausible, because "the tree method is based on a reductio ad absurdum strategy" (p. 75), that is, the assumption for the sake of argument of the *negation* of the conclusion to be proved. In fact, the tree method can be used to derive conclusions without using the reductio strategy (see Jeffrey 1981, Chapter 2).

Natural deduction relies on suppositions, which are sentences that are assumed for the sake of argument and which must be "discharged" if a derivation is to yield a conclusion. One way to discharge a supposition is to make it explicit in a conditional conclusion (conditional proof), and another way is to show that it leads to a contradiction and must therefore be false (reductio ad absurdum). An example is the following proof of an inference in the form known as *modus tollens*:

1. If the printer test was run then the printer produced a document.
2. The printer did not produce a document.
3. The printer test was run. (Supposition)
4. The printer produced a document. (Modus ponens applied to 1 and 3)

At this point, a contradiction occurs between one of the premises and the most recent conclusion. The rule of reductio ad absurdum discharges the supposition by negating it:

5. The printer test was not run.

Rips (1994) could have adopted a single rule for modus tollens, but it is a more difficult inference than modus ponens, so he assumes that it depends on the chain of inferential steps illustrated here. The main problems in developing a formal system are to ensure that it is computationally viable and that it explains robust psychological findings. An example of a computational difficulty is that the rule for introducing "and" can run amok, leading to such futile derivations as:

A
B
$\therefore$ A and B
$\therefore$ A and (A and B)
$\therefore$ A and (A and (A and B))

and so on ad infinitum. The rules that are dangerous are those that introduce a connective or a supposition. Programs in artificial intelligence, however, can use a rule in two ways: either to derive a step in a *forward* chain leading from the premises to the conclusion or to derive a step in a *backward* chain leading from the conclusion to the premises. In a backward chain, the effect of a rule is to create subgoals, for example, given the goal of proving a conclusion of the form, *A and B*, the rule for "and" creates a subgoal to prove *A* and a subgoal to prove *B*. If the program satisfies these two subgoals, then it has in effect proved the conjunction: *A and B*, and it terminates there with no further application of the rule. Rips (1994) prevents rules from running amok by using those that introduce connectives or suppositions only in backward chains. His system therefore has three sorts of rules: those that it uses forward, those that it uses backward, and those that it uses in either direction. Table 12.1 summarizes these rules in Rips's system.

The formal rules postulated in a psychological theory should be ones that naive individuals recognize as "intuitively sound" (Rips, 1994, p. 104). One worry about the rules in Table 12.1 is whether they are all intuitive. The rule for introducing "or," for example, was used appropriately by only 20% of participants in Rips's own study. Indeed, this rule is not part of other formal theories (e.g., Braine, 1978). What complicates matters is that Rips allows that individuals may differ in the rules they possess, they may learn new rules, and they may even use nonstandard rules that lead them to conclusions not sanctioned by classical logic (Rips, 1994, p. 103).

Table 12.1: The forward, backward, and bidirectional rules in Rips's (1994) system

Forward rules		
IF P THEN Q* P Q	IF P OR Q THEN R* P R	IF P AND Q THEN R P Q R
P AND Q* P	NOT (P AND Q)* (NOT P) OR (NOT Q)	NOT (P AND Q)* P NOT Q
P OR Q* NOT P Q	NOT (P OR Q) NOT P	
P OR Q IF P THEN R IF Q THEN R R	NOT NOT P* P	
Backward rules		
+P : Q IF P THEN Q	+NOT P : Q AND (NOT Q) P	+P : Q AND (NOT Q) NOT P
P Q P AND Q	P P OR Q	
P OR Q +P : R +Q : R R	NOT (P OR Q) (NOT P) AND (NOT Q)	

* Signifies that a rule can also be used backward. Rules, such as the one eliminating AND, are shown leading to the conclusion P; other versions of such rules yield the conclusion Q. Plus sign (+) designates a supposition and colon (:) designates a subsequent derivation.

A major problem for systems implementing proofs is to embody an efficient method of searching for the correct sequence of inferential steps. The process is computationally intractable, and the space of possible sequences of inferential steps grows very rapidly (Cook, 1971). Rips's system uses a fixed deterministic search procedure in evaluating an inference with a given conclusion. It tries each of its applicable forward rules in a breadth-first search until they yield no new conclusions. It checks whether the conclusion is among the results. If not, it tries to work backward from the conclusion,

pursuing a chain of inference depth first until it finds the sentences that satisfy the subgoals or until it has run out of rules to apply (Rips, 1994, p. 105). Either it succeeds in deriving the conclusion or else it returns to an earlier choice point in the chain and tries to satisfy an alternative subgoal. If all the subgoals fail, it gives up. However, Rips's system is incomplete, that is, there are valid inferences that it cannot prove. As Barwise (1993, p. 338) comments: "The 'search till you're exhausted' strategy gives one at best an educated, correct guess that something does not follow." In other words, when Rips's system fails to find a proof, it may do so because an inference is invalid or else because it is valid but the incomplete rules fail to yield its proof.

Rips's system constrains the use of suppositions. They can be made only in a backward chain of inference from a given conclusion, so reasoners can use suppositions only when there is a given conclusion or they can somehow guess a conclusion. In everyday life, reasoners are not constrained to making suppositions only when they have a conclusion in mind. "Suppose everyone suddenly became dyslexic," they say to themselves, and then they follow up the consequences to an unexpected conclusion, for example, the sale of dictionaries would decline. In an earlier account, Rips (1989) allowed suppositions to occur in forward chains of reasoning. But, in that case, how can they be prevented from running amok? One possibility is to distinguish between the strategies that reasoners adopt and the lower level mechanisms that sanction inferential steps. One strategy is to make a supposition, but the strategic machinery must keep the lower level mechanisms in check to prevent them from losing track of the purpose of the exercise. Indeed, human reasoners develop a variety of strategies for sentential reasoning, and they use suppositions in ways not always sanctioned by Rips's theory (van der Henst, Yang, & Johnson-Laird, 2002).

Braine and colleagues have described a series of theories based on natural deduction (see, e.g., Braine, 1978; Braine & O'Brien, 1998). Their rules differ from Rips's rules in two main ways. First, "and" and "or" can apply to any number of propositions, so they formulate the following rule to introduce "and":

$$\frac{P_1, P_2, \ldots P_n}{P_1 \text{ and } P_2 \ldots \text{and } P_n.}$$

Second, they do not distinguish between forward and backward rules. Instead, they try to build the effects of dangerous rules, such as: *P*; therefore, *P or* Q, into other rules. Hence, they have a rule of the form: *If* P_1 *or* $P_2, \ldots$ *or* P_n *then* Q; P_1; therefore, Q. Their idea is to obviate the need for the rule introducing disjunction. Like Rips, however, they appear to postulate a single deterministic search strategy in which individuals apply simple rules before they apply rules that make suppositions. A problem that both Rips and Braine share is that it is often not obvious what conclusion, if any, their theories predict that individuals should draw spontaneously from a set of premises. At this point, the first author should declare an interest. At one time, he was a proponent of formal rules of inference (see Johnson-Laird, 1975), but, as the next section illustrates, he has now come to believe that the human inference engine relies, not on form, but on content.

3. The Simulation of Spatial Reasoning Using Mental Models

The theory of mental models postulates that when individuals understand discourse, they construct models of the possibilities consistent with the discourse (e.g., Johnson-Laird & Byrne, 1991; Johnson-Laird, 2006). Each mental model represents a possibility. A frequent misunderstanding is that mental models are images. In fact, they are more akin to three-dimensional models of the world of the sort that underlie the phenomena of mental rotation (Metzler & Shepard 1982). Because each model represents a possibility, a conclusion is necessary if it holds in all the models of the premises, it is possible if it holds in at least one model of the premises,

and it is probable if it holds in most of the models of the premises given that the models are equiprobable. The theory accordingly embraces deductions, reasoning about possibilities, and probabilistic reasoning, at least of the sort that depends on the various ways in which events can occur (Johnson-Laird et al., 1999).

The first mental model theory was for simple inferences based on quantifiers, and programs have simulated various versions of this theory (see Bucciarelli & Johnson-Laird, 1999, for a review). Polk and Newell (1995) simulated a model theory in which counterexamples played no role, but more recent evidence implies that human reasoners do make use of them (Bucciarelli & Johnson-Laird, 1999; Johnson-Laird & Hasson, 2003). Bara, Bucciarelli, and Lombardo (2001) developed a program that simulated both sentential and quantified reasoning in a single model-based program. In contrast, Johnson-Laird has written a series of small-scale programs that simulate various sorts of reasoning. The general design of these programs is the same. Each program has a lexicon that specifies the meanings of the key words in the input, which, depending on the domain, may be sentential connectives, quantifiers, causal verbs, deontic verbs, relational terms, or nouns referring to objects. The program also has a grammar of the relevant fragment of English. In many cases, this fragment is infinite in size because the grammar contains recursive rules. Such a grammar is illustrated in the next section. Associated with each grammatical rule is a function that carries out the corresponding semantic interpretation. The parser is a "shift-and-reduce" one familiar in the design of compilers (see, e.g., Aho & Ullman, 1972). It constructs a representation of the meaning of each sentence as it uses the grammar to parse the sentence. The program accordingly implements a "compositional" semantics (Montague, 1974), that is, the meanings of the words in a sentence are composed to yield the meaning of the sentence from its grammatical structure. The resulting meaning can then be used to update the model, or models, of the discourse so far, which represent the context of each sentence. The present section illustrates how such a system works in a program for spatial reasoning.

The program simulates three-dimensional spatial reasoning based on mental models (Byrne & Johnson-Laird, 1989). The input to the program is a description with, or without, a given conclusion. There can be any number of premises, and they can describe complex three-dimensional relations. But a simple inference best shows how the program works:

The triangle is to the right of the circle.
The circle is to the right of the diamond.
Therefore, the triangle is to the right of the diamond.

The program composes a representation of the meaning of the first premise, which it uses to build a model. It uses the meaning of *the circle* to insert a token representing the circle into a minimal three-dimensional spatial model:

○

The meaning of *to the right of* specifies that the model-building system scans in a rightward direction from the circle, so the program increments the left-to-right axis from the circle while holding constant the values on the other two axes (up-and-down and front-and-back). It uses the meaning of *the triangle* to insert a representation of the triangle into an empty location in the model:

○ △

The left-to-right axis in this diagram corresponds to the left-to-right spatial axis of the model.

The program can search for referents in its spatial models. Hence, given the second premise:

The circle is to the right of the diamond

it discovers that the circle is already represented in its current model of the premises.

Table 12.2: Seven procedures for reasoning using models

1. Start a new model. The procedure inserts a new referent into the model according to a premise.
2. Update a model with a new referent in relation to an existing referent.
3. Update a model with a new property or relation.
4. Join two separate models into one according to a relation between referents in them.
5. Verify whether a proposition is true or false in models.
6. Search for a counterexample to refute a proposition. If the search fails, then the proposition follows validly from the previous propositions in the description.
7. Search for an example to make a proposition true. If the search fails, then the proposition is inconsistent with the previous propositions.

It uses the meaning of the sentence to update this model. It therefore inserts a representation of the diamond into an appropriate position in the model:

◇ ○ △

With the first premise, human reasoners can scan from the circle in the direction that the relation specifies to find a location for the triangle. But, with the second premise, this natural procedure is not feasible, because the subject of the sentence is already in the model. The program therefore scans in the opposite direction to the one that the relation specifies – from the circle to a location for the diamond. This task ought to be a little bit harder, and psychological evidence shows that it is (e.g., Oberauer & Wilhelm, 2000). If a premise refers to nothing in the current model, then the program constructs a new model. Later, given an appropriate premise, it can integrate the two separate models into a single model. This case also adds to the difficulty of human reasoning.

Given the putative conclusion in the example:

The triangle is to the right of the diamond

the program discovers that both referents are already represented in its current model. It checks whether the appropriate relation holds between them. It scans in a rightward direction from the diamond until it finds the triangle. The relation holds. Next, it checks whether any other model of the premises is a counterexample to the conclusion. It finds none, so it declares that the inference is valid. In case a conclusion does not hold in the current model, the program checks whether any other model of the previous premises allows the relation to hold. If not, the program declares that the proposition is inconsistent with what has gone before. Table 12.2 summarizes the main procedures used in the program. If the human inferential system uses models, it needs such procedures, too.

In formal systems, the previous inference can be proved only if an additional premise specifies the transitivity of "to the right of":

For any x, y, and z, if x is to the right of y, and y is to the right of z, then x is to the right of z.

This premise functions as an axiom for any inference concerning the relation, and for obvious reasons, logicians refer to such axioms as *meaning postulates*. Proof theory in logic and formal rule theories in psychology need meaning postulates to allow deductions whose validity depends on the meanings of relations. In contrast, as the program shows, the model theory does not need meaning postulates, because the validity of inferences emerges from the meanings of relations, which specify the direction in which to scan models, and from the procedures that construct models and search for counterexamples.

One point is easy to overlook. The program's search for counterexamples works because it has access to the representations of the *meanings* of the premises. Without

these representations, if the program were to change a model, it would have no way to check whether the result was still a model of the premises. Any inferential system that constructs alternative models therefore needs an independent record of the premises. It must either have a memory for their meanings, or be able to return to each premise to re-interpret it.

The strategy embodied in the spatial reasoning program is to construct a single model at a time. When a description is consistent with more than one layout, the program builds whichever model requires the least work. An alternative strategy, which is implemented in a program for reasoning about temporal relations, is to try to build all the different possible models. Still another strategy is to represent the alternative possibilities within a single model using a way to indicate the uncertain positions of entities in the model. Human reasoners can probably develop any of these strategies, depending on the particulars of the problems that they tackle (see, e.g., Carreiras & Santamaría, 1997; Jahn, Knauff, & Johnson-Laird, 2007; Schaeken, Johnson-Laird, & d'Ydewalle, 1996a, 1996b; Vandierendonck, Dierckx, & De Vooght, 2004).

The evidence corroborates the use of models in spatial reasoning. Participants in experiments report that they imagine layouts. They often make gestures with their hands that suggest they have a spatial model in mind. Likewise, if they have paper and pencil, they draw diagrams. Yet, such evidence does not rule out the possibility that deep down, the unconscious inferential processes are guided by form rather than content. Several experiments, however, provide crucial evidence supporting the model theory. One experiment used descriptions of two-dimensional spatial layouts of household objects and showed that inferences that depend on a single model are easier than those that depend on multiple models. Yet, the one-model problems called for longer formal proofs than the multiple-model problems (Byrne & Johnson-Laird, 1989).

A recent study demonstrated a still greater difficulty for meaning postulates (Goodwin & Johnson-Laird, 2005). It examined such inferences as:

Alice is a blood relative of Brian.
Brian is a blood relative of Charlie.
What follows?

The participants tended to infer that Alice is a blood relative of Charlie. They presumably thought of a set of siblings or a line of descendants. Yet, there are counterexamples to the conclusion. Suppose, for instance, that Alice is Brian's mother, and Charlie is his father. Alice is related to Brian, and he is related to Charlie, but his mother and father are probably not blood relatives. These "pseudo-transitive" inferences depend on relations that are neither transitive nor intransitive, but that yield models of typical situations in which a transitive conclusion holds. The model theory therefore predicts that the way to block these inferences is to get the participants to search harder for counterexamples. Hence, when the problem about "blood relatives" was prefaced with the clue that people can be related either by blood or by marriage, the proportion of transitive inferences was reduced reliably.

If human reasoners use formal rules to reason, then they need meaning postulates that capture the transitivity of relations. So what sorts of relations should be tagged as transitive? The reader might suppose that good candidates would be comparative relations, such as "taller than." But, consider this problem:

Cate is taller than Belle.
Belle *was* taller than Alice.
Who is tallest?

The change in tense no longer guarantees transitivity, and again individuals are much less inclined to draw the transitive conclusion (Goodwin & Johnson-Laird, 2005). It follows that no comparative terms, not even "taller than," can be classified as transitive in all cases. In other words, the logical form of an assertion depends on its significance, which in turn depends on its tense, its

context, and general knowledge. The obvious route to discover its correct logical form is to use this information to construct models of the situations to which it refers. But once one has constructed such models, they can be used directly in reasoning: There is no need to recover the assertion's logical form. Hence, if the system builds models, then it no longer needs meaning postulates. The models either support a transitive conclusion or not.

4. The Simulation of Sentential Reasoning Using Mental Models

Sentential reasoning hinges on negation and such connectives as "if," "or," and "and," which interconnect *atomic* propositions, that is, those that do not contain negation or connectives. Section 2 illustrated how sentential reasoning could be simulated using formal rules. Connectives have idealized meanings in logic, so that the truth-values of sentences formed with them depend solely on the truth-values of those atomic propositions or their negations that they interconnect. For example, an exclusive disjunction of the form: *A or else B but not both*, is true if one proposition is true and the other false, and in any other case the disjunction is false. Model theory in logic captures this analysis in a truth-table, as shown in Table 12.3. Each row in the table represents a different possibility, for example, the first row represents the case in which both *A* and *B* are true, and it shows that the disjunction is false in this case. Truth-tables can be used to determine the validity of sentential inferences: An inference is valid if any row in its truth-table in which the premises are true is also one in which its conclusion is true. However, truth-tables double in size with each additional atomic proposition in an inference, whereas the psychological difficulty of inferences does not increase at anything like the same rate (Osherson, 1974–1976).

The theory of mental models is based on a fundamental assumption that obviates this problem and that is known as the principle of *truth*:

Table 12.3: A truth-table for an exclusive disjunction

A	*B*	*A or else B but not both*
True	True	False
True	False	True
False	True	True
False	False	False

> Mental models represent only what is true, that is, they represent only true possibilities and within them they represent only those atomic propositions or their negations in the premises that are true.

As an example, consider an exclusive disjunction, such as:

> The machine does not work or else the setting is high, but not both.

The principle of truth implies that individuals envisage only the two true possibilities. They therefore construct the following two mental models shown in the rows of the following diagram, where "¬" designates negation:

```
¬ Machine works
                    Setting high
```

The principle of truth has a further, less obvious, consequence. When individuals think about the first possibility, they tend to neglect the fact that it is false that the setting is high in this case. Likewise, when they think about the second possibility, they tend to neglect the fact that it is false that the machine does not work in this case, that is, the machine *does* work. The relation between these mental models and the truth-table for an exclusive disjunction is transparent (see Table 12.3). The mental models correspond to those rows in the table in which the disjunction is true, and they represent only those *literals* in the premises that are true in the row, where a literal is an atomic proposition or its negation.

The principle of truth postulates that individuals normally represent what is true, but not what is false. It does not imply, however, that they never represent falsity. Indeed, the theory proposes that they represent what is false in "mental footnotes," but that these footnotes are ephemeral. People tend to forget them. But as long as they are remembered, they can be used to construct *fully explicit* models, which represent the true possibilities in a fully explicit way. Hence, the footnotes about what is false allow reasoners to flesh out the models of the proposition:

The machine does not work or else the setting is high, but not both.

to make them fully explicit:

¬ Machine works ¬ Setting high
Machine works Setting high

where a true negation is used to represent a false affirmative proposition. This representation of negation makes models more abstract than images, because you cannot form an image of negation. Even if you imagine, say, a large red cross superimposed on whatever is to be negated, nothing in the image alone captures the meaning of negation.

The meanings of conditional propositions, such as:

If the machine works then the setting is high

are a matter of controversy. Their meanings depend both on context and on the semantic relations, if any, between their two clauses – the *antecedent* clause following "if" and the *consequent* clause following "then" (see Johnson-Laird & Byrne, 2002). The core logical meaning of a conditional is independent of its context and of the meanings and referents of its antecedent and consequent clauses. It yields two mental models. One mental model represents the salient possibility in which both the antecedent and the consequent are true. The other model is wholly implicit, that is, it has no explicit content, but allows for possibilities in which the antecedent of the conditional is false. The mental models for the preceding conditional are accordingly:

Machine works Setting high
. . .

where the ellipsis denotes the implicit model, and a mental footnote indicates the falsity of the antecedent in the implicit possibilities. A biconditional, such as:

The machine works if and only if the setting is high

has exactly the same mental models, but a footnote indicates that both the antecedent and the consequent are false in the possibilities that the implicit model represents. It is the implicit model that distinguishes the models of a conditional from the model of a conjunction, such as:

The machine works and the setting is high

which has only a single model:

Machine works Setting high

The fully explicit models of the conditional can be constructed from the mental models and the footnote on the implicit model. They are as follows:

Machine works Setting high
¬ Machine works Setting high
¬ Machine works ¬ Setting high

Likewise, the fully explicit models of the biconditional are:

Machine works Setting high
¬ Machine works ¬ Setting high

One point bears emphasis: These diagrams refer to mental models, but mental models themselves represent entities in the world – they are not merely strings of words. Table 12.4 summarizes the mental models and the fully explicit models of sentences formed from the main sentential connectives in their "logical" senses.

Table 12.4: The mental models and fully explicit models for sentences based on the main sentential connectives

Connective	*Mental models*	*Fully explicit models*
A and B:	A B	A B
A or else B:	A	A ¬ B
	B	¬ A B
A or B, or both:	A	A ¬ B
	B	¬ A B
	A B	A B
If A then B:	A B	A B
	...	¬ A B
		¬ A ¬ B
If and only if A then B:	A B	A B
	...	¬ A ¬ B

"¬" denotes negation and "..." denotes a wholly implicit model. Each line represents a model of a possibility.

How are sentential inferences made with mental models? A computer program simulates the process (see Johnson-Laird & Byrne, 1991). The program takes as input a set of sentences. It is sensitive to the occurrence of the following sentential connectives: *and* (conjunction), *or* (inclusive disjunction), *ore* (exclusive disjunction), *if* (conditional), *iff* (biconditional), and *then*, which serves only a syntactic role.

The program has a grammar that can be summarized as follows, where the items in parentheses may, or may not, occur in a sentence, and *comma* is a syntactic element:

sentence = (negation) variable
= negation sentence
= (comma) sentence connective sentence
= (comma) *if* sentence *then* sentence.

These four rules allow for different sorts of sentences, but because "sentence" occurs on both the left- and right-hand sides of some rules, the rules can be used recursively to analyze complex sentences, such as:

if not A and B then, C or D

where *A*, *B*, C, and *D* are all variables. *Not A*, for example, is analyzed as a sentence according to the first rule in the set shown previously, and C *or D* is analyzed as a sentence according to the third rule. Each of the rules in the grammar has an associated function for carrying out the appropriate semantics, so that the parser controls the process of interpretation, too.

The program's process of inference can be illustrated by the following example:

A ore B.
Not A.
What follows?

The exclusive disjunction symbolized by "ore" yields the mental models:

A
B

The categorical premise yields the model:

¬ A

This model eliminates the first model of the disjunction because they cannot both be true. But it is consistent with the second model of the disjunction. Their conjunction:

¬ A B

yields the conclusion:

B.

This conclusion is valid, because it holds in all the models – in this case, the single model – consistent with the premises.

The principles for conjoining mental models seem straightforward, but contain some subtleties. If one model represents a proposition, *A*, among others, and another model represents its negation, ¬ *A*, their conjunction yields the empty (or null) model that represents contradictions. The previous

example illustrated this principle. But what happens if the two models to be conjoined contain nothing in common? An example illustrating this case occurs with these premises:

If C then D.
E ore C.

The reader is invited to consider what possibilities are compatible with the two premises. Most individuals think that there are two:

C D
E

The mental models of the first premise are:

C D
. . .

and the mental models of the second premise are:

E
C

One possibility according to the second premise is E, so the program conjoins:

C D and E

C occurs in the set of models of the disjunction from which E is drawn, so the interpretative system takes the absence of C in the model of E to mean *not* C:

C D and E ¬ C

Because there is now a contradiction – one model contains C and the other its negation – the result is the null model. The program next conjoins the pair:

C D and C

D does not occur elsewhere in the set of models of the disjunction containing C, so the two models are compatible with one another. Their conjunction yields:

C D

The program now constructs conjunctions with the implicit model of the conditional. The conjunction:

. . . and E

yields E, because E does not occur in the models of the conditional containing the implicit model. The final conjunction:

. . . and C

yields the null model, because C occurs in the models of the conditional, so its absence in the implicit model is treated as akin to its negation. The mental models of the conjunction of the premises are accordingly:

C D
E

The null models are not shown because they do not represent possibilities. The two models of possibilities yield the valid conclusion:

C and D, ore E.

Table 12.5 summarizes the mechanisms for forming conjunctions of pairs of models. These principles apply both to the combination of sets of models, as in the preceding disjunctive inference, but they also apply to the combination of possible individuals in models of quantified propositions (Johnson-Laird, 2006).

The same mechanisms apply to the conjunction of fully explicit models except that the first mechanism in the table does not come into play. Here are the previous premises again:

If C then D.
E ore C.

A mechanism that uses mental footnotes can flesh our mental models into fully explicit

Table 12.5: The mechanisms for forming conjunctions of pairs of mental models and pairs of fully explicit models

1. If one model represents a proposition, *A*, which is not represented in the second model, then if *A* occurs in at least one of the models from which the second model is drawn, then its absence in the second model is treated as its negation (and mechanism 2 applies); otherwise, its absence is treated as its affirmation (and mechanism 3 applies). This mechanism applies only to mental models.
2. The conjunction of a pair of models containing respectively a proposition and its negation yield the null model, e.g.:

   A B and ¬ A B yield nil.
3. The conjunction of a pair of models that are not contradictory yields a model representing all the propositions in the models, e.g.:

   A B and B C yield A B C
4. The conjunction of a null model with any model yields the null model, e.g.:
   A B and nil yield nil.

models. The fully explicit models of the conditional and the disjunction (see Table 12.4) are, respectively:

C	D	E	¬ C
¬ C	D	¬ E	C
¬ C	¬ D		

There are six pair-wise conjunctions, but three of them are contradictions yielding the null model. The remaining pairs yield the following models:

C	D	¬ E
¬ C	D	E
¬ C	¬ D	E

The same conclusion follows as before:

C and D, ore E.

But reasoners who rely on mental models will fail to think of the second of the these three possibilities.

A problem for formal rule theories is to find the right sequence of inferential steps to prove that a conclusion follows from the premises. The model-based program does not have a search problem, because it merely updates its set of models for each new premise. As the number of distinct atomic propositions in the premises increases, the number of models tends to increase, but it does so much less rapidly than the number of rows in a truth-table. Nevertheless, the intractability of sentential reasoning does catch up with the program and with human reasoners as the number of distinct atoms in a problem increases.

The principles for constructing conjunctions of mental models seem innocuous – just a slight variation on those for fully explicit models, which yield a complete account of sentential reasoning. After the program was written, however, it was given a test of the following sort of premises based on a hand of cards:

If there is a king then there is an ace ore if there is not a king then there is an ace.
There is a king.

When the program reasoned using mental models, it returned a single mental model:

King Ace

But when it reasoned using fully explicit models, it returned the fully explicit model:

King ¬ Ace

Did it really follow that there is *not* an ace? This result was so bizarre that Johnson-Laird

spent half a day searching for a bug in his program, but at last discovered it in his own mind. The force of the exclusive disjunction in the first premise is that one of the two conditionals is false, and the falsity of either conditional implies that there is not an ace, so the fully explicit models did yield a valid conclusion. Given an inclusive interpretation of the disjunction, or a biconditional interpretation of the conditionals, or both, mental models still yield the (invalid) conclusion that there is an ace, whereas fully explicit models do not. Nothing definite follows from the premises with these interpretations: There may, or may not, be an ace. Yet, as experiments showed (Johnson-Laird & Savary, 1999), nearly everyone succumbs to the illusion that there *is* an ace. Johnson-Laird modified the program so that it would search for illusions by generating a vast number of premises and comparing their mental models with their fully explicit models. Subsequent experiments corroborated the occurrence of various sorts of illusory inference in sentential reasoning (Walsh & Johnson-Laird, 2004), modal reasoning about what is possible (Goldvarg & Johnson-Laird, 2000), deontic reasoning about what is permissible (Bucciarelli & Johnson-Laird, 2005), reasoning about probabilities (Johnson-Laird et al., 1999), and reasoning with quantifiers (Yang & Johnson-Laird, 2000a, 2000b). The theories based on formal rules did not predict the illusory inferences, and they have no way of postdicting them unless they posit invalid rules of inference. But in that case, they then run the risk of inconsistency. Illusory inferences are therefore a crucial corroboration of the use of mental models in reasoning, and their discovery was a result of a simulation of the theory.

5. Concepts, Models, and Minimization

Because infinitely many valid conclusions follow from any set of premises, computer programs for proving theorems do not normally draw conclusions, but instead evaluate given conclusions (see, e.g., Pelletier, 1986). Human reasoners, however, exercise real intelligence because they can draw conclusions for themselves. They abide by two principal constraints (Johnson-Laird & Byrne, 1991). First, they do not normally throw semantic information away by adding disjunctive alternatives. Second, they aim for conclusions that re-express the semantic information in the premises parsimoniously. They never, for example, draw a conclusion that merely forms a conjunction of all the premises. Of course, human performance degrades with complex problems, but the goal of parsimony provides a rational solution to the problem of which conclusions intelligent programs should draw. They should express all the semantic information in the premises in a minimal description. The logic of negation, conjunction, and disjunction is often referred to as "Boolean," after the logician George Boole. Minimization accordingly has a two-fold importance. On the one hand, it is equivalent to the minimization of electronic circuits made up from Boolean units, which are powerful enough for the central processing units of computers (Brayton et al., 1984). On the other hand, cognitive scientists have argued that simplicity is a cognitive universal (Chater & Vitányi, 2003) and that the difficulty of the human learning of Boolean concepts depends on the length of their minimal descriptions (Feldman, 2000).

A simple algorithm to find a minimal description of a set of possibilities checks all possible descriptions, gradually increasing the number of literals and connectives in them, until it discovers one that describes the set. The problem is computationally intractable, and this method is grossly inefficient. Hence, various other methods exist (e.g., Quine, 1955), but, because of the intractability of the problem, circuit designers use approximations to minimal circuits (Brayton et al., 1984). Another version of the program described in the previous section uses the notation of the sentential calculus: & (conjunction), v (inclusive disjunction), ∇ (exclusive disjunction), $\rightarrow$ (conditional), and $\leftrightarrow$ (biconditional). It finds minimal descriptions using fully explicit

Table 12.6: The possibilities compatible with four Boolean concepts, putatively minimal descriptions of them, and true minimal descriptions discovered by the program using fully explicit models

III.	a ¬ b c ¬ a b ¬ c ¬ a ¬ b c ¬ a ¬ b ¬ c	
	Putative minimal description:	(¬ a & ¬ (b & c)) v (a & (¬ b & c))
	The program's description:	(¬ a v c) & (¬ b v ¬ c)
IV.	a ¬ b ¬ c ¬ a b ¬ c ¬ a ¬ b c ¬ a ¬ b ¬ c	
	Putative minimal description:	(¬ a & ¬ (b & c)) v (a & (¬ b & ¬ c))
	The program's description:	(c → (¬ a & ¬ b)) & (a → ¬ b)
V.	a b c ¬ a b ¬ c ¬ a ¬ b c ¬ a ¬ b ¬ c	
	Putative minimal description:	((¬ a & ¬ (b & c)) v (a & (b & c)))
	The program's description:	a ↔ (b & c)
VI.	a b ¬ c a ¬ b c ¬ a b c ¬ a ¬ b ¬ c	
	Putative minimal description:	(a & ((¬ b & c) v (b & ¬ c)) v ¬ a & ((¬ b & ¬ c) v (b & c))
	The program's description:	(a ∇ b) ↔ c

The Roman numbers are the labels of the problems in Shepard et al. (1961).

models. Table 12.6 presents four Boolean concepts, first studied by Shepard et al. (1961), with Feldman's (2000) putative minimal descriptions and, as the program revealed, actual minimal descriptions. Shepard et al. (1961) found that concepts III, IV, and V were roughly equally difficult for their participants to learn but VI was reliably harder, so Feldman concluded that subjective difficulty is well predicted by his putative descriptions. But, as the table shows, true minimal length does not correlate with psychological complexity. In fairness to Feldman, he used only approximations to minimal descriptions, and he restricted his vocabulary to negation, conjunction, and inclusive disjunction on the grounds that these are the traditional Boolean primitives. However, Goodwin (2006) has shown that when concepts concern patterns of switch positions that cause a light to come on, naive individuals neither restrict their vocabulary to these primitives nor are they able to discover minimal descriptions (less than 4% of their descriptions were minimal). Parsimonious descriptions are hard to find, and they may not relate to the psychological difficulty of learning concepts.

When the program builds models from premises, it multiplies them together to interpret conjunctions. Hence, to *describe* a given set of models, it works backward,

dividing the set up into subsets of models that can be multiplied together to get back to the original set. The process of division proceeds recursively until it reaches models that each contain only two items. Pairs of items are easy to describe, because the standard connectives do the job. Consider, for example, the following description (of concept V in Table 12.6):

$$((\neg a \,\&\, \neg (b \,\&\, c)) \vee (a \,\&\, (b \,\&\, c))).$$

It yields these fully explicit models of possibilities:

a	b	c
¬ a	b	¬ c
¬ a	¬ b	c
¬ a	¬ b	¬ c

The reader may notice, as the program does, that all four possible combinations of b and c, and their negations, occur in these possibilities. The program therefore recodes the models as:

a	X
¬ a	¬ X

where the value of the variable X is: b & c. The program compares these two models with each of its connectives and finds the description: a ↔ X. It plugs in the description of X to yield the overall minimal description: a ↔ (b & c).

There are six sorts of decomposition of a set of models depending on whether or not any pairs of propositions or variables occur in all four possible contingencies and how the other elements relate to them. Any procedure for minimization is necessarily intractable, but the program is more efficient than some algorithms. Table 12.7 presents some typical examples of its performance with examples from logic textbooks. Each example shows the input, and the program's output, which in each of these cases are both an evaluation of the given conclusion (the last assertion in the input) and a minimal valid conclusion expressing all the information in the premises.

6. General Discussion: The Nature of Human Deductive Reasoning

Does the engine of inference rely on form or content? Indeed, might it rely on entirely different principles? For example, Shastri and Ajjanagadde (1993) describe a "connectionist" system of simulating a network of idealized nerve cells capable of simple inferences (see also Chapter 2 in this volume on connectionist models). Likewise, in a series of striking studies, Oaksford and Chater (e.g., 1998) have argued that logic is irrelevant to our everyday reasoning and to our deductions in the psychological laboratory. The right criterion to assess them is the probability calculus, and they even claim that this calculus *describes* our deductions. Our reasoning, they say, is a rational adaptation to the world, and it depends on a tractable process. Because the probability calculus is not tractable, they argue that we use heuristics rather than the calculus itself. Yet, they reject Tversky and Kahneman's (e.g., 1973) findings that heuristics can lead to into errors in probabilistic thinking. They advocate instead different heuristics, which they claim are rational. Their theory is ingenious, and, granted certain assumptions about the interpretations of premises, it gives a good account of three sorts of reasoning: performance in a hypothesis-testing task, reasoning from simple quantified assertions, and simple inferences from conditional assertions (Oaksford & Chater, 1998). Yet, serious difficulties exist for the theses of rationality, tractability, and probability: Any heuristic process that seeks to circumvent the intractability of sentential reasoning is bound to lead to systematic error, for example, the illusory inferences described earlier. And recent empirical studies have shown that the probabilistic theory often makes unsatisfactory predictions (e.g., Espino et al., 2005; Schroyens, Schaeken, & d'Ydewalle, 2001). But the decisive issue is that the preservation of truth in a valid deduction transcends validity in the probability calculus (Rips, 2002).

Many observations suggest that human deductive reasoning does not rely on form.

Table 12.7: Some examples of the program using fully explicit models

1. Input: a v b
   a → c
   b → d
   c v d

Output: This premise follows from those that preceded it: c v d

Premises are equivalent to: (a → c) & ((b → d) & (a v b))

2. Input: w v (p → i)
   i → (c v s)
   s → u
   ¬ c & ¬ u
   ¬ w

Output: This premise follows from those that preceded it: ¬ w

Premises are equivalent to: ¬ w & (¬ p & (¬ i & (¬ c & (¬ s & ¬ u))))

3. Input: (a → b) & (c → d)
   (b → e) & (d → f)
   ¬(e & f)
   a → c
   ¬ a

Output: This premise follows from those that preceded it: ¬ a.

Premises are equivalent to: ¬ a & ((b → e) & ((e → ¬ f) & ((d → f) & (c → d))))

4. Input: (v → (s & (p & m)))
   (r → (g → (¬ l → ¬ m)))
   s → r
   (p → (c → g))
   l → ¬ c
   c
   ¬ v

Output: This premise follows from those that preceded it: ¬ v.

Premises are equivalent to: ¬ v & (¬ l & (c & ((p → g) & ((s → r) & ((m & g) → ¬ r)))))

One observation is that theorists have yet to devise an algorithm for recovering the logical form of propositions. Another observation is that the inferential properties of relations and connectives are impossible to capture in a simple way. Reasoners use their knowledge of meaning, reference, and the world to modulate their interpretation of these terms. Hence, no sentential connectives in everyday language, such as "if" and "or," can be treated as they are in logic. For example, the truth of a conjunction, such as, "He fell off his bicycle and he broke his leg," depends on more than the truth of its two clauses: The events must also be in the correct temporal order for the proposition to be true. Likewise, a conditional, such as "If she's in Brazil then she is not in Rio," has an interpretation that blocks a modus tollens inference (Johnson-Laird & Byrne, 2002), whereas a counterfactual conditional, such as "If she had been in Rio then she would have been in Brazil," facilitates the inference (Byrne, 2005). The use of axioms to specify the logical properties of relations, such as "taller than," faces similar problems. Logical properties depend on the proposition as a whole and its context. Instead, as the simulation program in Section 3 showed, reasoners can use the meanings of propositions

to construct appropriate models from which logical consequences emerge. A more recent simulation has shown how context, depending both on the current models of the discourse and on general knowledge, overrules the "logical" interpretations of connectives (Johnson-Laird, Girotto, & Legrenzi, 2004).

Because human working memory is limited in capacity, human reasoners cannot rely on truth-tables. Their mental models represent atomic propositions and their negations only when they are true in a possibility. The failure to represent what is false seems innocuous. Indeed, for several years, no one was aware of its serious consequences. However, the simulation program implementing the theory revealed for some inferences radical discrepancies between mental models and fully explicit models. These discrepancies predicted the occurrence of illusory inferences, which subsequent experiments corroborated. Some commentators argue that human reasoning depends on both formal rules and on mental models, and that the evidence shows only that sometimes human reasoners do not rely on logic, not that they never use formal rules. No conceivable evidence could ever rule out the use of formal rules on at least some occasions, but theoretical parsimony suggests that in general, human reasoners rely on mental models.

7. Conclusions

If humans err so much, how can they be rational enough to invent logic and mathematics, and science and technology? At the heart of human rationality are some simple principles that almost everyone recognizes: A conclusion must be the case if it holds in all the possibilities compatible with the premises. It does not follow from the premises if it runs into a counterexample, that is, a possibility that is consistent with the premises, but not with the conclusion. The foundation of rationality is our knowledge of these principles, and they are embodied in the programs simulating the theory of mental models.

References

Aho, A. V., & Ullman, J. D. (1972). *The theory of parsing, translation, and compiling, Vol. 1: Parsing.* Englewood Cliffs, NJ: Prentice Hall.

Bara, B., Bucciarelli M., & Lombardo V. (2001). Model theory of deduction: A unified computational approach. *Cognitive Science, 25,* 839–901.

Barwise, J. (1993). Everyday reasoning and logical inference. *Behavioral and Brain Sciences, 16,* 337–338.

Beth, E. W., & Piaget, J. (1966). *Mathematical epistemology and psychology.* Dordrecht, Netherlands: Reidel.

Braine, M. D. S. (1978). On the relation between the natural logic of reasoning and standard logic. *Psychological Review, 85,* 1–21.

Braine, M. D. S., & O'Brien, D. P. (Eds). (1998). *Mental logic.* Mahwah, NJ: Lawrence Erlbaum.

Brayton, R. K., Hachtel, G. D., McMullen, C. T., & Sangiovanni-Vincentelli, A. L. (1984). *Logic minimization algorithms for VLSI synthesis.* New York: Kluwer.

Bucciarelli, M., & Johnson-Laird, P. N. (1999). Strategies in syllogistic reasoning. *Cognitive Science, 23,* 247–303.

Bucciarelli, M., & Johnson-Laird, P. N. (2005). Naïve deontics: A theory of meaning, representation, and reasoning. *Cognitive Psychology, 50,* 159–193.

Byrne, R. M. J. (2005). *The rational imagination: How people create alternatives to reality.* Cambridge, MA: MIT Press.

Byrne, R. M. J., & Johnson-Laird, P. N. (1989). Spatial reasoning. *Journal of Memory and Language, 28,* 564–575.

Carreiras, M., & Santamaría, C. (1997). Reasoning about relations: Spatial and nonspatial problems. *Thinking & Reasoning, 3,* 191–208.

Chater, N., & Vitányi, P. (2003). Simplicity: A unifying principle in cognitive science? *Trends in Cognitive Science, 7,* 19–22.

Cook, S. A. (1971). The complexity of theorem proving procedures. In *Proceedings of the Third Annual Association of Computing Machinery Symposium on the Theory of Computing* (pp. 151–158).

Espino, O., Santamaría, C., Meseguer, E., & Carreiras, M. (2005). Early and late processes in syllogistic reasoning: Evidence from eye-movements. *Cognition, 98,* B1–B9.

Feldman, J. (2000). Minimization of Boolean complexity in human concept learning. *Nature, 407,* 630–633.

Goldvarg, Y., & Johnson-Laird, P. N. (2000). Illusions in modal reasoning. *Memory & Cognition, 28,* 282–294.

Goodwin, G. P. (2006). *How individuals learn simple Boolean systems and diagnose their faults.* Unpublished doctoral thesis, Princeton University, Princeton, NJ.

Goodwin, G., & Johnson-Laird, P. N. (2005). Reasoning about relations. *Psychological Review, 112,* 468–493.

Jahn, G., Knauff, M., & Johnson-Laird, P. N. (2007). Preferred mental models in reasoning about spatial relations. *Memory & Cognition,* in press.

Jeffrey, R. (1981). *Formal logic: Its scope and limits* (2nd ed.). New York: McGraw-Hill.

Johnson-Laird, P. N. (1975). Models of deduction. In Falmagne, R. J. (Ed.), *Reasoning: Representation and process in children and adults* (pp. 7–54). Hillsdale, NJ: Lawrence Erlbaum.

Johnson-Laird, P. N. (2006). *How we reason.* Oxford, UK: Oxford University Press.

Johnson-Laird, P. N., & Byrne, R. M. J. (1991). *Deduction.* Hillsdale, NJ: Lawrence Erlbaum.

Johnson-Laird, P. N., & Byrne, R. M. J. (2002). Conditionals: A theory of meaning, pragmatics, and inference. *Psychological Review, 109,* 646–678.

Johnson-Laird, P. N., Girotto, V., & Legrenzi, P. (2004). Reasoning from inconsistency to consistency. *Psychological Review, 111,* 640–661.

Johnson-Laird, P. N., & Hasson, U. (2003). Counterexamples in sentential reasoning. *Memory & Cognition, 31,* 1105–1113.

Johnson-Laird, P. N., Legrenzi, P., Girotto, V., Legrenzi, M., & Caverni, J-P. (1999) Naive probability: A mental model theory of extensional reasoning. *Psychological Review, 106,* 62–88.

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

Metzler, J., & Shepard, R. N. (1982). Transformational studies of the internal representations of three-dimensional objects. In Shepard, R. N., & Cooper, L. A., *Mental images and their transformations* (pp. 25–71). Cambridge, MA: MIT Press.

Montague, R. (1974). *Formal philosophy: Selected papers.* New Haven, CT: Yale University Press.

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

Oaksford, M., & Chater, N. (1998). *Rationality in an uncertain world.* Hove, UK: Psychology Press.

Oberauer, K., & Wilhelm, O. (2000). Effects of directionality in deductive reasoning: I. The comprehension of single relational premises. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 26,* 1702–1712.

Osherson, D. N. (1974–1976). *Logical abilities in children* (Vols. 1–4). Hillsdale, NJ: Lawrence Erlbaum.

Pelletier, F. J. (1986). Seventy-five problems for testing automatic theorem provers. *Journal of Automated Reasoning, 2,* 191–216.

Polk, T. A., & Newell, A. (1995). Deduction as verbal reasoning. *Psychological Review, 102,* 533–566.

Quine, W. V. O. (1955). A way to simplify truth functions, *American Mathematical Monthly, 59,* 521–531.

Rips, L. J. (1983). Cognitive processes in propositional reasoning. *Psychological Review, 90,* 38–71.

Rips, L. J. (1989). The psychology of knights and knaves. *Cognition, 31,* 85–116.

Rips, L. J. (1994). *The psychology of proof.* Cambridge, MA: MIT Press.

Rips, L. J. (2002). Reasoning. In Medin, D. (Ed.), *Stevens' handbook of experimental psychology, vol. 2: Memory and cognitive processes* (3rd ed., pp. 317–362). New York: John Wiley.

Schaeken, W. S., Johnson-Laird, P. N., & d'Ydewalle, G. (1996a). Mental models and temporal reasoning. *Cognition, 60,* 205–234.

Schaeken, W. S., Johnson-Laird, P. N., & d'Ydewalle, G. (1996b). Tense, aspect, and temporal reasoning. *Thinking and Reasoning, 2,* 309–327.

Schroyens, W., Schaeken, W., & d'Ydewalle, G. (2001). The processing of negations in conditional reasoning: A meta-analytical case study in mental models and/or mental logic theory. *Thinking & Reasoning, 7,* 121–172.

Shastri, L., & Ajjanagadde, V. (1993). From simple associations to systematic reasoning: A connectionist representation of rules, variables and dynamic bindings using temporal synchrony. *Behavioral and Brain Sciences, 16,* 417–494.

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

Tversky, A., & Kahneman, D. (1973). Availability: A heuristic for judging frequency and probability. *Cognitive Psychology, 5,* 207–232.

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

Vandierendonck, A., Dierckx, V., & De Vooght, G. (2004). Mental model construction in linear reasoning: Evidence for the construction of initial annotated models. *Quarterly Journal of Experimental Psychology, 57A,* 1369–1391.

Walsh, C., & Johnson-Laird, P. N. (2004). Coreference and reasoning. *Memory & Cognition, 32,* 96–106.

Yang, Y., & Johnson-Laird, P. N. (2000a). Illusory inferences with quantified assertions: How to make the impossible seem possible, and *vice versa*. *Memory & Cognition, 28,* 452–465.

Yang, Y., & Johnson-Laird, P. N. (2000b) How to eliminate illusions in quantified reasoning. *Memory & Cognition, 28,* 1050–1059.