The psychology of knights and knaves

LANCE J. RIPS*
University of Chicago

Received May 1988, final revision accepted July 1988

Abstract

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

Knight-knave brain teasers are about a realm in which some people, knights, *tell only truths, whereas all others,* knaves, *tell only lies. For example, suppose person A says, "I am a knight and B is a knight," and person B says, "A is a knave." Is A a knight or a knave? Is B a knight or a knave?*

In a pilot study, we asked subjects to think aloud while solving problems like these. Their statements suggested that they were making assumptions about the knight/knave status of the characters and drawing deductive inferences from these assumptions to test their consistency. This encouraged us to model the process by means of a simulation based on an earlier natural-deduction theory of reasoning. The model contains a set of deduction rules in the form of productions and a working memory that holds a proof of the correct answer. The greater the number of steps (assumptions and inferences) in the proof, the greater the predicted difficulty of the puzzle. The experiments reported here confirmed this prediction by showing that subjects were more likely to make mistakes (Experiment 1) and take longer to solve (Experiment 2) puzzles associated with a larger number of proof steps.

Introduction

Knight-knave puzzles begin like this: Suppose there is an island where there are just two sorts of inhabitants – *knights* who always tell the truth and *knaves* who always lie. Nothing distinguishes knights and knaves but their lying or truth-telling propensity. You overhear a conversation between two or more

*I thank David Cortner and Rita Walter for their help and advice on this project. NIMH Grant MH39633 supported the research. Experiment 1 was conducted at the University of Arizona with support from the Committee for Cognitive Science. Correspondence should be addressed to Lance Rips, Behavioral Sciences Department, University of Chicago, 5848 South University, Chicago, IL 60637, USA.

inhabitants, and on the basis of this conversation you must decide which of the individuals are knights and which are knaves. Smullyan (1978) is a rich source of these puzzles, of which the following is an example:

(1) We have three inhabitants, A, B, and C, each of whom is a knight or a knave. Two people are said to be of the *same type* if they are both knights or both knaves. A and B make the following statements:
A: B is a knave.
B: A and C are of the same type.
What is C? (Smullyan, 1978, p. 22.)

Although the answer isn't obvious, a little thought yields the solution. To get started, suppose that A is a knight. Since what he says is true, B would then be a knave. But if B is a knave, he is lying, which means that A and C are *not* of the same type. We're assuming that A is a knight; so on this assumption, C must be a knave. But what if A is a knave rather than a knight? Well, in that case, A's statement is false, and hence B is a knight. This makes A and C of the same type, which means that C is a knave. So no matter whether we take A to be a knight or a knave, C will be a knave, and this must be the answer to the puzzle.

Knight-knave puzzles are related to the well-known semantical paradoxes of logic and philosophy. For example, suppose an inhabitant, D, of Knight-knave Island says, "I am a knave." Is D a knight or a knave? It's easy to see that the assumption that D is a knight leads to the conclusion that D is a knave (if D is a knight, he must be telling the truth; hence, he is a knave since that is what he says he is). Similarly, the assumption that D is a knave entails that D is a knight (if D is a knave, he is lying; hence, he is a knight since that is the opposite of what he says). Paradoxes like these have profound implications for the theory of truth and have moved some philosophers (notably, Tarski, 1944) to view natural language as logically flawed (see Kripke, 1975, for a current interpretation of the liar paradoxes that is less inimical to natural language). But there is nothing paradoxical about puzzles like (1), and their significance for our purpose lies in what they reveal about the nature of human reasoning.

This paper describes a model of how ordinary subjects – ones without special training in logic – deal with knight-knave problems. The model is tested in two experiments, one focusing on subjects' success rate in solving fairly difficult puzzles, the other on response times for easier examples. As it turns out, success rates and response times can both be explained on the hypothesis that subjects try to solve the problems by applying the mental equivalent of simple deduction rules, such as modus ponens (*IF p THEN q* and *p* implies *q*), conjunction elimination (*p AND q* implies *p*), and a few others.

Research in the psychology of deductive reasoning has been limited to a few specific paradigms. Indeed, much of the literature has focused on just two kinds of tasks: evaluating classical syllogisms and solving the selection puzzle (for reviews, see Evans, 1982, and Wason & Johnson-Laird, 1972). But these problems are highly restricted; they involve only a small subset of potential arguments and don't generalize easily to deductive arguments of other types. To determine whether our present theories are able to handle the full range of humanly possible deductions, we need a richer sampling of logical formats. Obviously, knight-knave problems have their own limitations, since they all depend on the basic definitions of *knight* and *knave*. Still, there are an infinite number of such problems, and the level of reasoning required to solve them scales a wide range, as we will see. Thus, they may provide a better window on general inference processes than some other popular paradigms. The goal, though, is not to promote knight-knave experiments over classical-syllogism or selection-task experiments, but to expand the scope of investigation in this area.

Protocol evidence

As a preliminary attempt to find out how people handle such problems, I asked a group of subjects to solve four of them and to think aloud as they did so. The four problems were drawn from Smullyan (1978), with slight rewordings to clarify the task. Puzzle (1) was among this group. Each problem was typed on an index card that subjects could inspect at any time; however, they were unable to write down any notes of their own. The four subjects were University of Chicago undergraduates who had not taken a formal course in logic. Each subject tried to solve all four problems in an order determined by a random Latin square. Subjects' remarks were tape-recorded and later transcribed.

Table 1 presents a complete transcript from one of the subjects, a college freshman, who was working on Problem (1). In general, her line of attack follows the pattern we gave earlier. The subject begins by assuming that person A is a knight. Since what A says is true on this assumption and since A says that B is a knave, the subject infers that B is lying (line b). B's statement that A and C are of the same type must therefore be false. But by assumption, A is a knight, and thus C must be a knave. So by line d of Table 1 the subject is able to conclude that B and C are knaves if A is a knight, and she calls this her "first possibility." She then turns to the second possibility: that A is a knave. This means that B is a knight, so that A and C are of the same type, namely knaves. In line g, though, the subject runs into a temporary problem in that she has forgotten C's "type" under the first possi-

Table 1. *Protocol from subject 3 on knight-knave problem (1) (Ellipses indicate pauses.)*

a. Well, the *type* introduction, that's a little confusing.
b. If A is a knight, then B has to be lying.
c. So A and C are of, are not of the same type.
d. That would mean that B and C are knaves.
e. So that's one possibility.
f. If A is a knave, then B is a knight, and A and C are both knaves.
g. Now I've forgotten my first possibility.
h. If A is a knight, then B ... if A is a knight, B is a knave and C is a knave.
i. So my first possibility is C is a knave.
j. Second possibility is if A is a knave ... Wait a minute ...
k. If A is a knight, no, if A is a knight, then C is a knave.
l. Uh, if A is a knave ..., then C is a knave.
m. So either way, C is a knave.

bility. This was not uncommon in the experiment, since subjects were not able to write down their intermediate results. In lines h and i, she goes back to re-compute the first part of her solution, and in line j she begins again on the second part. But before she develops the latter possibility, she apparently has some second thoughts about the first. Finally, she reminds herself of the implications of the second possibility, and in line m correctly concludes that on either possibility C is a knave.

The protocol in Table 1 is one of the most articulate solutions in our sample, but some of the properties of this answer are representative. First, the subjects attempted to solve the problems by considering specific assumptions. In order to solve Problem (1), the subject of Table 1 tries to determine whether the speakers, A and B, are knights or knaves, since she can then use the truth or falsity of their statements to determine whether C is a knight or knave. However, the problem doesn't identify A or B directly, and so it is not clear how to get a start on the answer. The subject's strategy is to make an assumption or supposition about A's status (that A is a knight) and see what this supposition implies about C. Once she has determined that in this case C is a knave, she can back up and make the opposite assumption that A is a knave. The transcripts contain many similar instances of assumption-making.

Second, subjects tended to work forward from their assumptions about the lying or truth-telling of the speakers to implications for the question. They usually didn't make assumptions about the answer and work backward to see if they are implied or contradicted by the given information, even though this strategy is equally logical. For example, another way to solve (1) is to show that the assumption that C is a knight leads to a contradiction so that (by reductio ad absurdum) C must be a knave. There is no evidence in the transcripts that the subjects attempted such a strategy on this problem. In other problems there are hints of backward reasoning, but they are rare. Along the same lines, subjects ordinarily use the fact that a particular individual is a knight or knave to establish the truth or falsity of what that individual says, rather than going from the truth or falsity of a statement to the status of the speaker. We should exercise caution here, however, since lack of evidence for backward reasoning may be due to difficulties subjects have in describing it in the thinking-aloud context.

Finally, subjects usually had the logical resources they needed to solve the puzzles, but sometimes forgot assumptions, confused intermediate results, or gave up too soon. For example, one of the subjects began her attack on Puzzle (1) like this:

> A says B is a knave, that's either true or false. Keeping that in mind, B says that A and C are of the same type. So if A is telling the truth, C is also of A's type, which is truth-telling – knights – A and C are both knights if B is telling the truth. If B is telling the truth and A is telling the truth, well, something, neither, not both of them can be right, because either A is correct about B's being a knave, or ... wait, this is getting confusing ...

This subject tries to consider all possible ways in which A and B could be assigned to the knight and knave categories and begins to get lost in the process. There are cases in which subjects do run up against more clearly logical troubles, but most of the subjects' difficulties involved conceptual bookkeeping rather than narrowly logical deficiencies.

Although this protocol evidence is partly determined by the specific problems and conditions of the experiment, there may be something more general in subjects' strategy of making assumptions and working forward from them. The protocols suggest that subjects are using a particular type of deductive reasoning, one that is substantially different from a strategy based, for example, on truth tables or semantic tableaux (Beth, 1955). The following section describes a simulation model for these problems that attempts to capture subjects' strategy. The model predicts the relative difficulty of knight-knave puzzles in terms of the number of steps required for their solution, and these predictions are put to the test in the following experiments.

A computational model for knight-knave puzzles

The proposed model derives from a prior theory of human propositional reasoning (Rips, 1983, 1984), which is based on the idea that people deal with deduction problems by applying mental-deduction rules, like those of formal natural-deduction systems (Gentzen, 1969; Jaskowski, 1934). The theory is therefore similar to earlier psychological proposals by Braine (1978) and Osherson (1974–1976), which also rely on natural-deduction frameworks. The deduction rules apply to information stored in working memory and perform inferences that follow from it. A typical deduction rule, for example, is And Elimination, which applies to a working-memory sentence of the form *p AND q* and produces the two sentences *p* and *q*, stated separately. The deduction rules implement elementary inference principles; but by stringing these inferences together, people can create more complex mental derivations or proofs that show how a remote conclusion follows from its premises. The theory predicts subjects' performance on a deduction problem in terms of the length of the required derivation and the availability of the rules: The shorter the derivation and the more available the rules that generate it, the faster and more accurate subjects should be.

The model for knight-knave problems is a version of the natural deduction system with a few additional rules to handle the special constraints of the task. To represent the new rules, we use the expressions *knight(x)* to mean that *x* is a knight, *knave(x)* to mean *x* is a knave, and *says(x,p)* to mean that person *x* uttered the sentence *p*. So, for example, *says(A, knave(B))* represents the proposition that A said B is a knave. In these terms, the four new rules are the ones in Table 2. The first allows us to infer that *p* is true if a knight said it; the second, that *p* is false if a knave said it; the third, that someone who is not a knave is a knight; and the fourth, that someone who is not a knight is a knave. Other rules could be obtained from the problem

Table 2. *Knight-knave rules used in constructing problems for Experiments 1 and 2*

Rule 1:
says(x, p) and *knight(x)* entail *p*.

Rule 2:
says(x, p) and *knave(x)* entail *NOT p*.

Rule 3:
NOT knave(x) entails *knight(x)*.

Rule 4:
NOT knight(x) entails *knave(x)*.

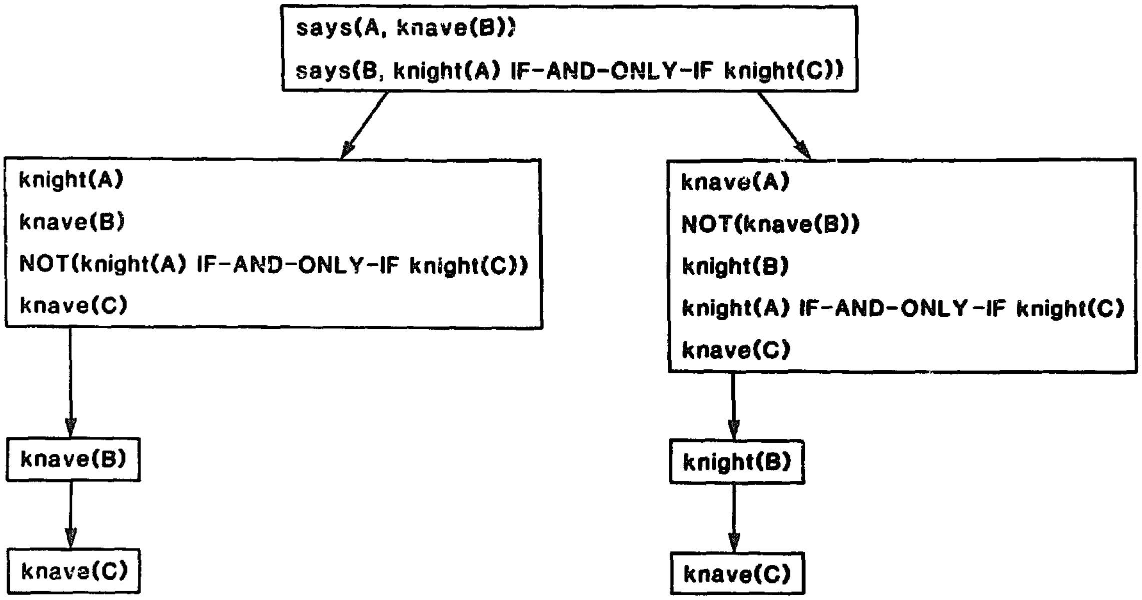
definition; for example, it is also true that *says(x, p)* and *NOT p* entail *knave(x)*. But, as mentioned earlier, these inference patterns were not very common in the protocols and were therefore not included in the model. The remaining rules are all simple inferences from propositional logic, which depend on the sentence connectives NOT, AND, OR, or IF. It is possible, of course, to construct knight-knave puzzles that depend on more complex logic; however, these propositional rules are sufficient to create problems that span a wide range of difficulty and enable us to test the model's basic features.

The model exists as a PROLOG program that accepts sentences of the form just described and makes assumptions and draws inferences about knight/knave identity.[1] The program consists of a simple production system linked to representations in working-memory. These representations include the assumed and deduced sentences, together with the dependency relations among them. In the latter respect, the model resembles the AI reasoning systems of Stallman and Sussman (1977) and Doyle (1980). The program begins by storing the (logical form of the) sentences in the problem and extracting from them the names of the individuals (e.g., A, B, and C). It then assumes that the first-mentioned individual – usually, A – is a knight and draws as many inferences as it can from this assumption and the given sentences. The program obtains the inferences by applying its rules to the stored sentences, initially in the order given in Table 2. If the program detects a pair of contradictory sentences (e.g., *knight(B)* and *knave(B)*) during this process, it immediately abandons its assumption that A is a knight and assumes instead that A is a knave. However, if the new set of inferences is consistent, it proceeds to assume that the second-mentioned individual is a knight. After each step, the program revises the ordering of its rules so that rules that have successfully applied will be tried first on the next round. The program continues in this way until it has found all consistent sets of assumptions about the knight/knave status of the individuals. Finally, it reports that an individual *x* is a knight if *knight(x)* appears in all of the consistent sets, that *x* is a knave if *knave(x)* appears in all of the consistent sets, and that *x*'s identity is undetermined in all other cases.

As an example of the program's operation, let's consider how it would

[1] Readers who know PROLOG may find this use of the language odd, since the model is in effect a theorem prover built on top of a language that contains its own theorem-proving mechanism (see, e.g., Clocksin & Mellish, 1981). Why not take advantage of PROLOG's native logical abilities to solve the problems directly? The answer is that the model attempts to specify the cognitive processes of human novices, and these processes are probably far removed from PROLOG's own sophisticated resolution methods. For this reason, PROLOG functions here simply as a convenient programming language, just as if we had used LISP. Using a logic-based programming language to construct a model of human reasoning is no stranger than the fact that AI reasoning systems (including PROLOG, for that matter) run on hardware that has its own logic circuitry.

Figure 1. *Working memory representation from the model's solution to Puzzle (1). See text for explanation.*


says(A, knave(B))
says(B, knight(A) IF-AND-ONLY-IF knight(C))
knight(A)
knave(B)
NOT(knight(A) IF-AND-ONLY-IF knight(C))
knave(C)
knave(A)
NOT(knave(B))
knight(B)
knight(A) IF-AND-ONLY-IF knight(C)
knave(C)
knave(B)
knave(C)
knight(B)
knave(C)

solve Problem (1). Since the program has no facility for parsing English sentences, the given information must be presented in logical form. The first given sentence is simply *says(A, knave(B))*. The second can be represented as *says(B, knight(A) IF-AND-ONLY-IF knight(C))*, since we can capture the notion of *same type* as a biconditional. The program stores these sentences in a node of its working memory, as shown at the top of Figure 1. None of the program's inference rules apply to these sentences directly. To make any headway, the program has to try out some assumptions, and the first assumption it makes is *knight(A)*. This assumption and the inferences that follow from it are stored in a subordinate node of working memory in order to indicate their hypothetical status (see Rips, 1983). The new assumption, together with the first of the original sentences, triggers one of the program's knight-knave rules (Rule 1 of Table 2), which permits it to infer *knave(B)*. This in turn yields *NOT (knight(A) IF-AND-ONLY-IF knight(C))* by knight-knave Rule 2. The program then uses a propositional rule to deduce *knave(C)* from the negated biconditional and the assumption *knight(A)*. At this point, no more inferences follow. The program briefly considers whether B might be a knight, but rejects this possibility immediately since it directly contradicts the conclusion that B is a knave. It then stores the assumption that B is knave at a subordinate node in memory.[2] Similarly, it rejects the possibility that C is a knight in favor of the assumption that C is a knave.

The program has now found a consistent set of assumptions: A is a knight and B and C are knaves. However, it is not through, since it has yet to consider the possibility that A is a knave. It therefore backs up and explores the consequences of this assumption, as shown on the right-hand side of Figure 1. From *knave(A)*, the program can conclude that *NOT(knave(B))* and hence *knight(B)*, according to Rules 2 and 3. This implies *knight(A) IF-AND-ONLY-IF knight(C)* by Rule 1. One of the propositional rules recognizes that the biconditional and the assumption *knave(A)* yields the final conclusion *knave(C)*. Thus, the only assumptions about B and C that are consistent with the possibility that A is a knave are that B is a knight and C a knave, and these appear in the bottom right nodes of the figure. This means the program has found two consistent sets of assumptions: Either A is a knight and B and C are knaves or B is a knight and A and C are knaves. Because the identity of A and B depends on the assumptions, the program describes them as uncertain. But it declares C a knave, since this is true in both sets. This solution follows, in outline, the method used by the subject of Table 1. The two branches of the memory tree in the figure correspond to the "two possibilities" that she discusses.

The model just described is in some ways simpler than the theory of propositional reasoning on which it is based. First, all rules in the present model operate in a "forward" (or bottom-up) direction from the given information toward the conclusion. The parent theory (Rips, 1983) contains inference rules that operate in the reverse direction, from the main goal or conclusion to potential subgoals. Backward rules are omitted from the knight-knave model since the protocols showed little evidence of subgoaling, as we noted earlier. This may in turn reflect the fact that these knight-knave problems call for a decision about the identity of the characters, where the decision can be reached by breaking down the given information. Backward reasoning is likely to be more common when the conclusion itself is complex and must be built-up from components. Second, we assume in what follows that subjects' error rates and response times depend only on the length of the derivation, that is, on the number of inferences needed for a correct answer, and not on the difficulty of applying specific rules. Although the latter factor is important in general, we designed the stimuli in these experiments to minimize its effect.

[2] The bottom nodes in Figure 1 are redundant since the information they contain has already been deduced and since these nodes cannot give rise to any new inferences. They are included in the program mainly for the sake of uniformity. Although it would be easy to eliminate them, the small savings in working memory capacity would be offset by an increase in the complexity of the program.

This has the advantage of allowing us to test the model without having to estimate free parameters.

Experiment 1: Prediction of solution rates

To test the model, we constructed a variety of knight-knave puzzles that could be solved by means of the knight-knave rules in Table 2, together with the propositional rules of Table 3. Previous cognitive theories have claimed the propositional rules of the table as psychologically primitive. For example, Rules 5, 6, 9, and 11 appear in the theory of Braine, Reiser, and Rumain (1984); Rules 5, 7, 8, and 11 in Osherson (1975), and Rules 5, 6, 8, and 9 in Rips (1983). Rule 10 is the only one that has not appeared in previous models, but it seems an obvious corollary of Rule 9. We claim that these rules are the elementary inference principles that subjects will rely on in solving the puzzles in our stimulus ensemble. We do not assume, however, that the Table 3 rules necessarily exhaust the primitive inferences that subjects are able to draw.

To derive predictions about the difficulty of the problems, we submitted them to the PROLOG program described earlier and counted the number of inference steps that the program needed to solve them. This inference-step

Table 3. *Propositional rules used in constructing problems for Experiments 1 and 2*

Rule 5 (AND Elimination):
p AND q entails *p, q.*

Rule 6 (Modus Ponens):
IF p THEN q and *p* entail *q.*

Rule 7 (DeMorgan–1):
NOT (p OR q) entails *NOT p AND NOT q.*

Rule 8 (DeMorgan–2):
NOT (p AND q) entails *NOT p OR NOT q.*

Rule 9 (Disjunctive Syllogism–1):
p OR q and *NOT p* entail *q.*
p OR q and *NOT q* entail *p.*

Rule 10 (Disjunctive Syllogism–2):
NOT p OR q and *p* entail *q.*
p OR NOT q and *q* entail *p.*

Rule 11 (Double Negation Elimination):
NOT NOT p entails *p.*

measure serves as our main independent variable. However, the problems also varied in the number of knight or knave characters (either 2 or 3) and in the number of clauses in the problem statement. We therefore paired the problems so that the two items in each pair contained the same number of individuals and clauses, but differed in the number of steps in their solutions. Our basic prediction, then, is that, within a given pair, the problem with a larger number of inferences will produce larger error rates.

Method

The subjects in this experiment received a group of knight-knave problems, and they decided for each person in a problem whether that person was a knight, a knave, or was undetermined. At the beginning of the experiment, we gave subjects a detailed introduction to the type of puzzle they would see. We illustrated the definitions of knight and knave with sentences that might be said by a knight (e.g., *A says, "2 + 2 = 4"*) or a knave (e.g., *B says, "2 + 2 =5"*). A sample problem showed them how to mark their answer sheets, which listed each of the speakers alongside boxes labeled "knight," "knave," and "impossible to tell." We read these instructions to the subjects, while they followed along in their own booklets. We then gave subjects a packet containing 34 problems, one problem per page. (The problems appeared in a different random order for each subject.) They proceeded through the booklet, under instructions to work the problems in order and not to return to a problem once they had completed it. Although we recorded the approximate amount of time they spent on the task, the subjects worked at their own pace. Unlike the subjects of the pilot experiment, these subjects were able to write down any information they wished.

Problems

The experimental problems consisted of a list of speakers (A, B, and C) and their utterances, and they required subjects to mark the type of each speaker or to mark "impossible to tell." Six of the problems had two speakers; the remaining 28 had three. The two-speaker problems contained three or four clauses, while the three-speaker problems contained four, five, or nine clauses. For these purposes, a clause is an elementary phrase such as *B is a knave* or *I am not a knight.* We counted clauses in terms of the underlying form of the sentence; so both *A is a knave and B is a knave* and *A and B are knaves* contain two clauses. A sentence such as *All of us are knights* counts as two clauses – i.e., *knight(A)* and *knight(B)* – in the context of a problem with two speakers and as three clauses in a three-speaker problem. As we mentioned, problems were paired in order to equate the number of speakers and clauses. As an example, Problems (2) and (3) formed one of the pairs of four-clause, three-speaker items:

(2) A says, "C is a knave."
B says, "C is a knave."
C says, "A is a knight and B is a knave."
(3) A says, "B is a knight."
B says, "C is a knave or A is a knight."
C says, "A is a knight."

We had initially planned to use all 34 problems in our statistical comparison, but subsequently found that the program was unable to solve one of them. This problem depended on a propositional rule that was not in Table 3. Since no predictions were available for this item, we omitted it in the analyses that follow, along with the item with which it was paired. For the remaining problems, the simulation program printed a complete derivation, according to the principles discussed in the preceding section. We calculated the predicted difficulty of the problem as the number of assumed and inferred propositions that the program considered before reaching the correct answer. For example, the program needed 16 steps to solve Problem (2), and 20 steps for Problem (3). Hence we predict that subjects will make more mistakes on (3) than on (2).

Subjects

Thirty-four subjects participated in this study in three groups of 10 to 13 individuals. All subjects were University of Arizona undergraduates, native English speakers who had never taken a course in formal logic. For taking part in this experiment, they received experimental credit in their introductory psychology course, and in order to encourage them to perform accurately, we also paid them for correct responses. The instructions told subjects they would start with a $3.40 bonus, but with 10 cents subtracted for each problem they missed. They were also told that the minimum payment would be 50 cents.

Despite this incentive, 10 of the subjects stopped working the problems within 15 minutes of the beginning of the test and scored at slightly less than chance accuracy: The percentage of problems answered correctly by this group was 2.5%; chance accuracy was 5.1%. These *peripheral* subjects (as we will call them) were apparently unwilling or unable to engage in the task, and we will therefore report the results separately for the entire set of subjects and for just the remaining *core* subjects.

Results and discussion

Solution rate was 20%, with a range from 0% correct for the least successful subject to 84% correct for the most successful one. Although this overall score is low, it clearly exceeds the chance level of 5% accuracy just mentioned. For core subjects, the rate increased to 26%, with the same range. The individual problems varied over a more modest interval: None of the subjects solved the most difficult problem and 35% solved the easiest one.

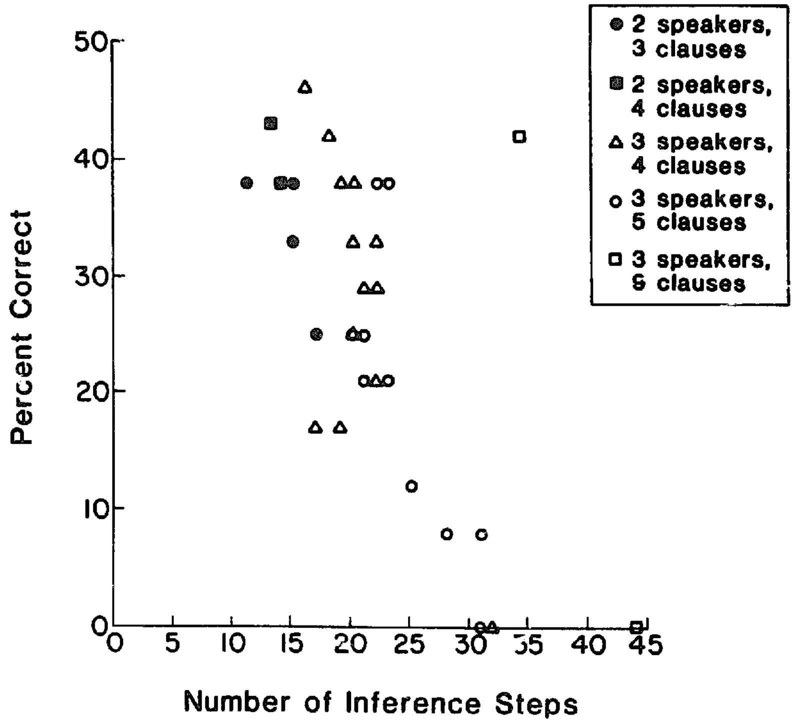
These data support the model's basic prediction concerning the relative difficulty of paired items. Subjects solved 24% of the problems that the model predicted to be easier, but 16% of the problems the model predicted to be difficult. This difference is significant when problem pairs serve as the unit of analysis ($t(15) = 2.50, p = .025$), and also when subjects serve as the unit ($t(33) = 3.71, p < .001$). In absolute terms, the difference is fairly small, but the low overall solution rate puts a cap on the size of the effect. Moreover, there is only a small theoretical difference in the number of steps that the two groups of problems require. The simulation used a mean of 19.3 steps in solving the simpler problems and 24.2 steps in solving the harder ones. The difference between the two types of problems widens slightly if we consider the core subjects alone. This group solved 32% of the easier problems and 20% of the more difficult ones, an effect that is again significant over pairs ($t(15) = 2.89, p = .011$) and over subjects ($t(23) = 4.66, p < .001$).

We can get a more fine-grained view of the inference effect by plotting the percentage of correct responses for each problem against the predicted number of steps. This plot appears in Figure 2, with the problems broken down according to the number of speakers and clauses. The data here are from the core subjects, but the pattern from the entire group is very similar. Notice, first, that although there are some deviations, these data exhibit the predicted downward trend within a given class of problem.

Second, the residual effects of speakers and number of clauses are relatively small. Figure 2 shows that two-speaker problems tend to be easier than three-speaker problems, but the distribution of scores for the latter completely overlaps that of the former. If we consider just the four-clause puzzles, which provide the clearest comparison, we find a 40% solution rate with two speakers, and a 28% rate with three. However, this difference is only marginally significant ($t(23) = 1.72, p = .10$), possibly because of the small number of two-speaker items. (Comparable figures for the entire group of subjects are 29% vs. 21%, $t(33) = 1.51, p = .14$.)

There is no consistent trend for the number of clauses. On two-speaker puzzles, core subjects scored 33% correct with three clauses and 40% correct with four. For three-speaker puzzles, core subjects were 28% correct with

Figure 2. *Percentage of correct solutions in Experiment 1 as a function of the number of inference steps used by the model.*



four clauses, 20% correct with five, and 21% correct with nine. The data were no clearer for the full set of subjects.

Third, the plot shows that one of the three-speaker, nine-clause problems is an outlier, displaced to the right of the other problem types. It thus appears to be easier than one would expect on the basis of the number of steps it requires (relative to the remaining problems). This item is Problem (4), which the model solves in 34 steps and which was answered correctly by 42% of core subjects (29% of all subjects):

(4) A says, "We're all knaves."
B says, "A, B, or C is a knight."
C says, "A, B, or C is a knave."

One reason why (4) may be easier than we anticipated has to do with the logical form of the sentences. Since the rules in Table 3 treat the connectives AND and OR as binary, each of the sentences in the problem has to contain two connectives. For example, the underlying form of the first sentence is *says(A, ((knave(A) AND knave(B)) AND knave (C))*. If A is a knight, this implies *((knave(A) AND knave(B)) AND knave(C))*, and it takes the model two more steps (via Rule 5) to unpack this into its separate clauses.

Our binary formulation may make this problem more difficult for the program than it was for the subjects. Braine (1978) and McCawley (1981) have proposed that the *and* and *or* of natural language are *n*-ary, not binary, connectives. On this approach, we should represent the inner part of the first sentence as something like *AND(knave(A), knave(B), knave(C))* instead of the form shown earlier. By recasting Rule 5 to take advantage of this alternative representation, we can then decompose the sentence in one step rather than two. (The revised rule would simply say that from a conjunction of the form $AND(p_1, p_2, \ldots, p_k)$ we can infer each of $p_1, p_2, \ldots, p_k$.) We can similarly generalize Rules 7–10, thereby reducing the number of steps involved in dealing with the second and third sentences of (4). Hand simulation shows that the revised rules can solve this problem in 27 steps, which brings it into better agreement with the rest of the stimuli. There are, of course, other ways to explain this discrepancy. For instance, subjects could be solving the problem by means of a strategy or a rule that does not exist in the program's repertoire. Unfortunately, there are too few problems of this longer variety to allow us to sort out these possibilities. There was only one other nine-clause problem, and no subject managed to solve it.

In sum, the findings are consistent with the basic prediction that subjects should score higher on puzzles with a smaller number of inference steps. Some uncertainties arise when we try to compare problems across stimulus pairs, particularly with respect to the nine-clause problems. Within pairs, however, we observe a small but statistically reliable difference that lends support to the model. Obtaining a more precise test would obviously entail some changes in design. For one thing, the problems were apparently quite difficult for most of these subjects, and the high error rate limits our ability to detect subtle effects. For another, the pairing of the puzzles was fairly coarse. Although we equated the members in a pair for the number of speakers and underlying clauses, there was no control over other potentially important factors. For example, the correct response varied within pairs, as did the connectives that figured in the problem statement. In the following experiment, we attempt to control all of these factors while still maintaining a difference in the number of inferences.

Experiment 2: Prediction of response times

The goal of this experiment is to provide a more stringent test of the natural-deduction model. In the first place, we try to show that the model is able to predict the amount of time subjects take to reach a correct solution. The form of the prediction is analogous to what we have seen before: The more steps the model needs to find the answer to a problem, the longer subjects should take to get it right. The response-time measure, however, motivated us to simplify the problems. Puzzles such as (1)–(4) would produce extremely long and variable times and would yield too few correct answers for analysis.

Second, we attempted to impose tighter control on the form of the problems in order to avoid the confoundings discussed in the previous section. To see how this can be done, consider the following puzzles:

(5) A: "I am a knave and B is a knave."
B: "I am a knight."
(6) A: "I am a knave and B is a knave."
B: "A is a knave."
(7) A: "I am a knight and B is a knight."
B: "A is a knave."

Notice that all three items have exactly the same surface and underlying form, differing only in the content of their clauses. In particular, the only connective in these problems, the *and* in the first sentence, is constant across (5)–(7). The problems also have the same answer, since in each of them A must be a knave and B a knight. Nevertheless, the model predicts Problem (7) to be more difficult than either (5) or (6). One reason for this is that in (5) and (6) the model quickly disposes of the (incorrect) possibility that A is a knight. For if A is a knight in these first two problems, then what A says is true, which is that he and B are knaves. But this means that A himself is a knave, contrary to assumption. By contrast, if the A of (7) is a knight, we're entitled to conclude from his statement only that he and B are knights. We must consult B's statement and realize that if B is a knight then A must be a knave, before we can rule out the possibility that A is a knight. Thus, (7) will require more steps in total than either (5) or (6). We take advantage of matched triples such as (5)–(7) in this experiment to eliminate irrelevant effects of problem wording and response.

The natural-deduction model for these problems was essentially the same as the one we considered earlier. However, we made two minor modifications to allow the simulation to solve a slightly wider variety of puzzles. This change concerned Rules 9 and 10 in Table 3, rules that implement the so-called Disjunctive Syllogism. We supplemented these rules so that the program would infer *p* from any of the following combinations of sentences: (a) *OR(knight(x), p)* and *knave(x)*; (b) *OR(knave(x), p)* and *knight(x)*; (c) *OR(p, knight(x))* and *knave(x)*; and (d) *OR(p, knave(x))* and *knight(x)*.

Method

Subjects in this study viewed the problems on a monitor and decided for each problem about the knight/knave status of its two characters. The subjects' instructions were similar to those of Experiment 1, with the addition of information about the trial sequence and the response apparatus. The subjects controlled a response panel that contained a single button at the left and three clustered buttons at the right. At the start of the trial, they were to have their left index finger on the left-hand button and their right index finger on the center button of the three-button group. The monitor signaled the beginning of the trial with the word "START," and when the subjects were ready to begin, they pressed the left button. The screen cleared, and then presented the problem in the form shown in (5)–(7), with the prompt "A?" underneath. At this point, subjects decided on the identity of person A and pressed one of the outer two buttons at the right with their right index finger to indicate their answer. After they did so, they returned their finger to the middle and pressed the center button. The prompt "B?" appeared on the screen, and the subjects made one last button press in the same way to record their decision about person B. Finally, the monitor presented the subjects with feedback about the accuracy of their answer and the amount of time they had taken.

The instructions told subjects that some of the problems would be difficult and that they should be sure their decision was correct before responding. They were also told, "once you have found the right answer, don't delay in pressing the button. Respond to each problem as fast as you can without making any errors." To give them some practice with the procedure, the computer took them through 12 practice trials in which they saw the names A and B followed by the word "knight" or "knave." The subjects responded by pressing the buttons corresponding to the presented words. For half the subjects the "knight" button was at the right of center and the "knave" button at the left, while for the remaining subjects this assignment was reversed. A microcomputer randomized the problems in a new order for each subject, controlled the trial sequence, and recorded the button presses and response times.

Problems

We chose the problems from those formed by selecting all possible combinations of options in the following frame:

$$\text{(8)}\quad \text{A: “A is}\begin{bmatrix}\text{a knight}\\ \text{a knave}\\ \\ \text{not a knight}\\ \text{not a knave}\end{bmatrix}\begin{bmatrix}\text{and}\\ \\ \text{or}\end{bmatrix}\text{B is}\begin{bmatrix}\text{a knight}\\ \text{a knave}\\ \\ \text{not a knight}\\ \text{not a knave}\end{bmatrix}\text{.”}$$

$$\text{B: “}\begin{bmatrix}\text{A}\\ \\ \text{B}\end{bmatrix}\text{is}\begin{bmatrix}\text{a knight}\\ \text{a knave}\\ \\ \text{not a knight}\\ \text{not a knave}\end{bmatrix}\text{.”}$$

This procedure yields 256 potential problems. However, since *NOT(knight(x))* is equivalent to *knave(x)* and *NOT(knave(x))* to *knight(x)*, we can think of this set as consisting of 32 problem groups, where the 8 problems within each group differ only in the presence of negatives. For example, Problems (9) and (10) are from the same group:

(9) A: “A is a knave and B is a knave.”
B: “B is a knight.”
(10) A: “A is not a knight and B is a knave.”
B: “B is not a knave.”

To make the sentences less awkward, we replaced the names with the pronoun *I* where it was appropriate. For example, instead of the sentence *B: “B is a knight,”* subjects saw *B: “I am a knight.”*

We submitted each of the problems to the natural-deduction program and chose 12 of the groups for this experiment on the basis of the output. The sample problems in Table 4 summarize the selection. Each of the sample problems in the table represents one problem group, as defined earlier. The rows of the table show that three groups of problems had *A is a knight* and *B is a knight* as the correct response; three had *A is a knight* and *B is a knave*; three had *A is a knave* and *B is a knight*; and three had *A is a knave* and *B is a knave*. The three groups within each row also had the same connective (*and* or *or*). However, the groups differ in the number of inference steps that the model needed to solve them: The problems in the first two columns required relatively few steps (13.1 for column 1 and 13.0 for column 2) and those in the last column relatively many (16.4 steps). In what follows, we refer to items in column 1 (which contain the sentence *B: “I am a knight”* or *B: “I am not a knave”*) as the *first type* of *small-step* problems and to items in column 2 as the *second type* of *small-step* problems. Column 3 contains *large-step* problems. Our prediction, then, is that response times should be longer and errors more frequent for the large-step problems within each row of the table.

Subjects received a total of 96 problems (12 groups x 8 problems per group). However, because of a programming error, four of the problems contained mistakes (substitution of *knight* for *knave* or the opposite error) in the form in which the subjects saw them. Two of these problems contained two negatives and two contained three negatives. For this reason, we will consider only the data from the 48 problems with zero or one negative in the results below.

Subjects

Fifty-three University of Chicago undergraduates took part in this experiment. They had answered an advertisement in the University newspaper and were paid $4.00 for their time. Like the subjects of the previous experiment, all were native speakers of English and none had taken a formal logic course. In addition to their base pay, they also received a bonus for accuracy: a $5 maximum minus 10 cents per trial on which they made an error. On the basis of the earlier study, we expected that many subjects would be unable to complete the test without making a large number of incorrect responses.

Table 4. *Sample problems from Experiment 2, with correct response and relative number of inference steps*

	Number of inference steps		
Correct response	Small		Large
	Type 1	Type 2	

A = Knight
B = Knight
	A: "I am a knave or B is a knight."
	B: "I am a knight."
		A: "I am a knave or B is a knight."
		B: "A is a knight."
			A: "I am a knight or B is a knave."
			B. "A is a knight."

A = Knight
B = Knave
	A: "I am a knave or B is a knave."
	B: "I am a knight."
		A: "I am a knave or B is a knave."
		B: "A is a knave."
			A: "I am a knight or B is a knight."
			B: "A is a knave."

A = Knave
B = Knight
	A: "I am a knave and B is a knave."
	B: "I am a knight."
		A: "I am a knave and B is a knave."
		B: "A is a knave."
			A: "I am a knight and B is a knight."
			B: "A is a knave."

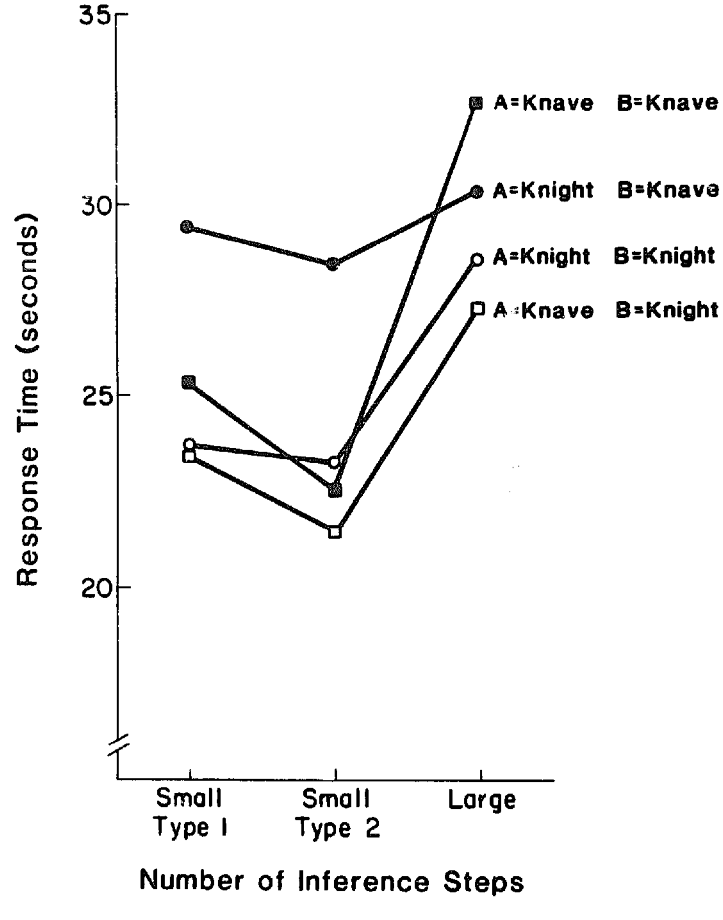
A = Knave
B = Knave
	A: "I am a knave and B is a knight."
	B: "I am a knight."
		A: "I am a knave and B is a knight."
		B: "A is a knight."
			A: "I am a knight and B is a knave."
			B: "A is a knight."

Since these trials are useless for measuring response time, we decided at the outset to discard data ırom those subjects who made errors on more than 40% of trials. Thirty subjects from the group succeeded in making fewer errors than this cut off.

Results and discussion

Figure 3 summarizes the mean correct response times and shows that problems with a larger number of predicted inference steps took longer to solve. The solution times that we have plotted in this figure measure the interval

Figure 3. *Mean correct response times from Experiment 2 as a function of the number of inference steps used by the model. The individual lines indicate the correct response to the different classes of problem: A is a knight and B is a knight (open circles), A is a knight and B is a knave (filled circles), A is a knave and B is a knight (open squares), and A is a knave and B is a knave (filled squares).*



from the point at which the problem appeared on the screen to subjects' button press for the *second* character of the problem. The resulting mean times are arranged in the figure to follow the organization of Table 4. Each curve in the figure indicates a particular response combination.

The critical result is the effect of inference steps: On average, subjects took 25.5 and 23.9 s to solve the two types of small-step problems, but 29.5 s on the large-step problems. To examine this effect, we performed an analysis of variance of the solution times and then calculated a contrast between the large-step and small-step items. In this analysis, we replaced missing observations due to errors with the mean of the remaining times for the relevant condition. The contrast proved reliable, with $F(1,58) = 24.66, p < .001$. The orthogonal contrast between the two small-step problem types is, however, nonsignificant, $F(1,58) = 1.95$, $p > .10$. This is precisely the pattern we would expect if subjects were solving the problems in the way the model does. To check the relation between response times and errors, we counted a trial as incorrect if a subject misidentified either character in the problem. The error rate was 15.8% for the first type of small-step problem, 9.0% for the second type, and 14.4% for the large-step problems. We had expected that large-step items would yield higher error rates than either of the small-step types. This relationship holds for small-step problems of the second type and reverses by 1.4 percentage points for the first. It seems highly unlikely that a reversal of this size could cause a speed–accuracy trade off that would compromise the large difference in solution times.

Figure 3 also shows that the times depended on the response that the problem demanded. Subjects took an average of 24.8 s to solve problems whose correct answer was *knight(A)–knight(B)*, 29.4 s for *knight(A)–knave(B)*, 24.0 s for *knave(A)–knight(B)*, and 26.8 s for *knave(A)–knave(B)*. The difference among these means is reliable according to the analysis of variance just mentioned, $F(3,87) = 3.55$, $p = .018$. The error rates also indicated that the *knight(A)–knave(B)* problems were the most difficult combination. In particular, error rates were 14.4% for *knight(A)–knight(B)*, 17.5% for *knight(A)–knave(B)*, 8.0% for *knave(A)–knight(B)*, and 12.2% for *knave(A)–knave(B)*. The difficulty of the *knight(A)–knave(B)* problems is not easy to explain. There is, in fact, a difference in the number of inference steps that might account for it: The model used 14.4 steps on average for *knight(A)–knight(B)* puzzles, 14.6 steps for *knight(A)–knave(B)*, 13.3 steps for *knave(A)–knight(B)*, and 14.3 steps for *knave(A)–knave(B)*. However, this variation in steps is so small that it seems unreasonable to think that it fully explains the solution times. Nor is the presence of a disjunction in the *knight(A)–knave(B)* problems the decisive factor. If disjunctions were especially difficult, we would also expect longer times and higher error rates for the *knight(A)-knight(B)* puzzles, contrary to the results in Figure 3.

Most of the increase in times for the *knight(A)–knave(B)* response is due to the small step items; so we should look to these items for clues. Some subjects may have thought (incorrectly) that if a character says "I am a knave ..." then the character must be lying, no matter how the sentence continues, and that otherwise the character is telling the truth. For the small-step problems, this would lead these subjects to respond that A is a knave and that B is a knight, a response that is just opposite to that required by the difficult *knight(A)–knave(B)* items. The tendency would also result in correct answers (for the wrong reasons) to the *knave(A)–knight(B)* problems, and these problems do in fact have the lowest error rates of the four response classes. It is possible that even when subjects respond correctly to the *knight(A)–knave(B)* problems, they take longer to do so because of this competing strategy; if so, this would serve to explain the response-time difference. The main difficulty with this explanation is that it predicts an interaction between response category and the number of steps in the problem. None of the characters in the large-step problems say *I am a knave* (see Table 4); so by the simple strategy just outlined, subjects should respond that both A and B are knights. This would lead to longer times for *knave(A)–knave(B)* problems and shorter ones for *knight(A)–knight(B)*. There is a hint of this cross-over in Figure 3, but the interaction is not significant, $F < 1$.

A final factor worth mentioning is the effect of negatives. There are two reasons to anticipate that negatives would produce longer solution times: First, subjects may take longer to read and comprehend explicitly negative sentences (e.g., Clark & Chase, 1972; Wason & Johnson-Laird, 1972); second, the model stipulates that extra steps are necessary to transform these negatives to positives (e.g., to transform *NOT(knave(x))* to *knight(x)* via Rule 3). In line with this, subjects took 23.4 s to solve problems with no negatives and 27.2 s when there was one negative, $F(1,87) = 8.20$, $p < .01$. The error rates show a similar pattern with 10.6% errors for problems with no negatives and 13.9% for problems with one negative. However, the negation effect did not interact with the difference between small- and large-step problems, nor with the effect of response category, $F < 1$ in both cases. The three-way interaction of step size, response category, and negation was also nonsignificant, $F < 1$.

General discussion

According to the natural-deduction model, people carry out deduction tasks by constructing mental proofs. They represent the problem information, make further assumptions, draw inferences, and come to conclusions on the basis of this derivation. This process is supposed to be a general one, applying to all humanly solvable deduction tasks, but most previous evidence for it has come from experiments in which subjects judge the validity of arguments (Braine et al., 1984; Osherson, 1974–1976; Rips, 1983). The present pair of studies extends the model to a new domain: a set of knight-knave puzzles that depend on logical properties but that don't possess an explicit premise–conclusion format and don't call for judgments of validity. In this domain the model performs quite well. When subjects thought aloud while solving the puzzles, their statements followed the assume-and-deduce strategy that typifies natural-deduction proofs. Experiment 1 showed that a specific implementation of the model could predict the probability of subjects solving a set of moderately complex and varied puzzles. Experiment 2 generalized this result, demonstrating that response times increased with the number of steps in the underlying proof.

In interpreting these results, however, we need to recognize some limitations, both in the nature of the data and its relation to the theory. First, although most subjects could deal with at least the simpler problems, a large minority apparently found these problems extremely difficult, performing at chance levels. Since the theory has little to say about why this is so, this group raises questions about the scope or boundary conditions for the model. Second, we have interpreted the results only in the natural-deduction framework and have so far neglected other approaches. Since there is, in fact, no consensus about the nature of deductive reasoning (or even whether people are *capable* of such reasoning), we should examine the findings from other points of view. Perhaps some of the alternatives are equally able to account for the findings. We take up these issues in the remainder of the paper.

Subjects who did not complete the task

In one respect, it's not too surprising that some subjects failed to grapple with the problems; for knight-knave problems wouldn't be puzzles if they weren't somewhat difficult. But why the large variation among subjects? Why were some subjects in Experiment 1 able to achieve scores of over 80% correct whereas others missed all of the problems? Since the model, as we have so far described it, has no place for individual differences, we are faced with the prospect of a theory that applies to only a select subset, leaving unsuccessful subjects out of account. Of course, it would be easy to explain away the differences as effects of motivation. If such an explanation were correct, individual differences would leave the theory unscathed, since the model can't be blamed if subjects don't take the problems seriously. We have no evidence at this time about the truth of such an explanation; examining it would mean collecting data from subjects on a variety of problem types and checking whether knight-knave puzzles produce a differential pattern of scores. In the absence of such psychometric data, it's at least worth contemplating the possibility that there's something special about these puzzles that can cause trouble for even motivated subjects.

Individual differences have appeared previously in tasks where subjects evaluate the validity of arguments, and we have tried to explain such results in terms of the availability (or pragmatic acceptability) of specific deduction rules (Rips & Conrad, 1983). For example, one rule that appears to cause differences is OR Introduction, which states that a sentence *p* entails *p OR q* for arbitrary *q*. While some subjects routinely accept arguments whose validity depends on OR Introduction (according to the natural-deduction model), other subjects just as routinely reject them. The model can accommodate this by treating the availability of OR Introduction as a parameter varying across subjects.

It is unclear, however, whether the same device could explain individual differences in the present task. Availability of the propositional rules in Table 3 is unlikely to account for them, since we deliberately avoided puzzles that depend on controversial rules such as OR Introduction. Even if some of the rules did vary in availability, we would expect subjects' performance to suffer on just the subset of problems for which those rules were crucial. We wouldn't expect the blanket failure that some of the subjects experienced. A more likely culprit, perhaps, is the availability of the knight-knave rules, particularly Rules 1 and 2 of Table 2, which figure in all of the problems. Certainly, if the subjects don't understand that what a knight says is true and what a knave says is false, then they won't be able to deal with the task at all. This amounts to the suggestion that some subjects simply didn't comprehend or weren't able to carry out the instructions we gave them. Although we have little direct evidence about this possibility, what we have is mixed. On one hand, the comments of a few subjects at the conclusion of Experiment 1 did suggest this kind of misunderstanding. On the other, none of the protocols from the pilot subjects expressed anything like this kind of mistake. Perhaps this mixed evidence reflects a difference in subject population, but we cannot be sure.

A final possibility has to do with the way subjects organized the problem-solving process. Some of the protocols suggest that at the start of the session some subjects don't have the systematic strategies that they develop on subsequent trials. The second of the two protocols quoted in the Introduction to this paper provides an example of this type of initial difficulty. It may be, then, that the rules in Tables 2 and 3 are equally available to everyone, but that subjects differ in the ease with which they hit upon a stable solution path. Our model for these puzzles does not deal with this start up phenomenon, since it has no mechanism for learning which strategy to apply. We envision a natural-deduction system that could bootstrap its way to a strategy, given a specification of the task (Rips, 1988), but such a system is probably a long way off.

It would not be surprising if all of the factors just discussed were involved in subjects' failure. Initial difficulty in understanding the task or in finding a workable strategy could quickly discourage those whose motivation was already low. Understanding the exact reasons for subjects' difficulties may reveal some gaps in the model; but there are no obvious reasons to think that these difficulties reflect a serious flaw in the model's central tenets. The claim is that the basic natural-deduction framework is common to all deduction tasks, however it happens to be applied in specific cases.

Alternative theories

Deduction by heuristic

Although the present experiments do not directly test rival theories of deductive reasoning, they may still have implications for this debate. Consider, for example, the thesis that people are incapable of deduction outside a narrow range that excludes even simple conditional inferences (Evans, 1982; Pollard, 1982). Although the high error rates that we have just discussed lend some support to this view, many of the subjects did reasonably well. We have already noticed that the most successful subjects scored over 80% in Experiment 1. In Experiment 2, one of our subjects made no errors at all on the 48 critical problems, and 15 additional subjects scored at least 90% correct. Of course, we pre-selected the subjects in the second experiment for accurate responding, so it is hardly surprising to discover that they did well. However, accurate performance on the part of even a few subjects is something that a theory of reasoning must explain. Explaining errors in reasoning is usually easy since – as we have just seen – there are multiple ways in which they can occur. Correct performance is what taxes a theory, since it suggests more elaborate and more interesting mental processes.

It is possible, certainly, that the successful subjects were right for the wrong reasons – that they based their correct responses on simple heuristics that fell short of true deductive reasoning. But which heuristics? In discussing Experiment 2, we mentioned one possibility in connection with the effect of response categories. This was a tendency to respond "knave" if the relevant character said *I am a knave* ... (or *I am not a knight* ...) and to respond "knight" otherwise. However, a glance at Table 4 shows that strict adherence to this heuristic would have produced scores no higher than 25% correct, whereas the obtained rate was 87%. Moreover, this heuristic probably reflects a partial logical insight, since it may well be due to subjects recognizing that neither knights nor knaves can utter the isolated sentence *I am a knave*. Finally, even if some unknown, non-deductive heuristic could account for the results of the two experiments, it would be hard-pressed to explain protocols such as the one quoted in Table 1. This subject is clearly in control of the deductive properties of the problem, including implications both of the key terms (i.e., *knight*, *knave*, and *type*) and of the conditionals that she constructs in lines b, f, h, and k. There are no apparent "non-logical" short cuts but, instead, a step-by-step analysis.

Deduction by pragmatic schemas

The deficiencies of the heuristic approach also plague more recent theories that tie reasoning to "pragmatic" domains, such as permission-giving (Cheng & Holyoak, 1985; Holland, Holyoak, Nisbett, & Thagard, 1986). A basic problem is that the island of knights and knaves seems as remote from the pragmatic world as it is possible to be. "Pragmatic reasoning," in the intended sense, means that reasoning is a function of schemas that are shaped by everyday circumstance. But a situation in which people always tell the truth or always lie is probably not one that many of us have had experience with. Of course, lying and truth-telling are common enough, and we probably do have schemas for dealing with them. Tarski's famous biconditional truth schema – "p" is true IF AND ONLY IF p – may be one such example (and is related to Rules 1 and 2 in Table 2). But the mere understanding of lying and truth-telling is insufficient to solve the present problems without further knowledge of the properties of the connectives AND and OR. The results of the experiments strongly suggest that at least the successful subjects do have knowledge of these properties that is independent of any obvious pragmatic domain.

The basic evidence for the pragmatic-schemas view comes from experiments showing improved performance in Wason's selection task when the problem is phrased in terms of permissions or restrictions, but not in terms of conditionals that express unfamiliar relations (Cheng & Holyoak, 1985; Griggs & Cox, 1982; Wason & Green, 1984). Although Cheng and Holyoak acknowledge that their "findings need not be interpreted as evidence against the very possibility" of natural deduction, they hold that "people typically reason using schematic knowledge that can be distinguished from ... context-free syntactic inference rules" (Cheng & Holyoak, 1985, p. 409) and that rules at the level of natural logic "are probably only rarely applied to semantically meaningful material" (Holland et al., 1986, p. 282). However, it is a big step from their results to the latter conclusions. By analogy, it is also true that people don't swim effectively with their feet encased in cement and that their performance improves dramatically when the cement is removed. But it doesn't follow from this that people typically swim by virtue of being freed from cement. To show that "people typically reason using schematic knowledge," one would have to make the case that people use these schemas on most deduction problems, not just that schemas are helpful in the selection task.[3]

Deduction by mental models

Finally, consider the proposal by Johnson-Laird (1983) that when people reason deductively they do so by constructing mental models of the content of the problem. On this approach, reasoning begins when a subject sets up an internal diagrammatic model of a situation that is consistent with the given facts of the problem. The subject then surveys the model for a potential conclusion and, if one is found, attempts to find a counterexample to the conclusion by altering the model. If no counterexample appears, the subject adopts the initial conclusion as correct. If there is a counterexample, the first conclusion is rejected and another conclusion examined. This process continues until the subject reaches an acceptable conclusion or decides that no conclusion is valid. Of the alternative theories that we have reviewed, this one comes closest to giving the flavor of the reasoning that the subject of Table 1 engaged in. In this transcript, as well as in the others that we collected, subjects adopt definite hypotheses about the knight-knave status of the characters, where these hypotheses may constitute representations for specific states of affairs.

The difficulty with mental models comes in fleshing out the details of the problem-solving process. For example, what would be a mental-models approach to Problem (1) (which we repeat here for reference)?

(11) We have three inhabitants, A, B, and C, each of whom is a knight or a knave. Two people are said to be of the *same type* if they are both

[3] We note that another issue surrounding pragmatic schemas is whether inferences based on them can't be alternatively explained in terms of mental rules analogous to those in modal logic. Proponents of pragmatic reasoning deny that schema-based inferences can be captured by purely "syntactic" rules. However, modal calculuses exist for the very concepts that the pragmatic schemas are supposed to explicate (see, e.g., Føllesdal & Hilpinen, 1971, and Lewis, 1974, on logics for permission and obligation; and Lewis, 1973, for causality). It is possible that a version of these rules is psychologically real (Osherson, 1976, presents some evidence on this point). Supporters of pragmatic reasoning have yet to discuss the relation between modal logic and their own schemas.

knights or both knaves. A and B make the following statements:

A: B is a knave.
B: A and C are of the same type.

What is C?

As one possibility, a subject might begin by constructing a mental diagram containing a token for A labeled "knight" to stand for the possibility that A is a knight. Since his statement is true in this model, B must be a knave; so we must add a token for B labeled "knave." This means that B's statement is false, and hence A and C are of different types We must therefore add a third token for C that also has the "knave" tag. At this point, then, our mental model would look something like this:

(12) knight_A
knave_B
knave_C

From this representation, we can read off the tentative conclusion that C is a knave.

We must now ask whether there are other models that are consistent with the given information but in which C is not a knave. To check this, we can attempt to construct a model in which A is labeled "knave." Then B is a knight; so this token must also be changed. Since B's statement is now true, A and C are of the same type; hence token C again gets the "knave" label. The resulting model looks like this:

(13) knave_A
knight_B
knave_C

But since C is still a knave in this model, our initial conclusion stands. The correct answer is that C is a knave.

How compelling is this account of Problem (1)? Certainly, the "models" in (12) and (13) conform to the possibilities that the subject in Table 1 contemplates; so the theory has some initial plausibility. It's worth recognizing that neither this subject nor any of the other pilot subjects mentioned envisioning or manipulating a situation with tokens corresponding to A, B, and C. But perhaps this can be put down to some difficulty in describing such models. The real trouble is that the theory provides no account of the process that produces and evaluates these models. For example, consider the step that results in adding knight_B to the model in (13). The most obvious way to explain this step is to say that we recognize that if A is a knave, his statement is false; that is, it is not the case that B is a knave. We also recognize that if it is not the case that B is a knave, then B is a knight. Johnson-Laird (1983) explicitly denies that this is due to mental inference rules or meaning postulates such as those in Tables 2 and 3. He also denies that this kind of reasoning is simply a matter of non-logical heuristics. But assuming this is true, what cognitive mechanism achieves these insights?

Another possibility is that models such as (12) and (13) are put together in a more haphazard way, then checked for consistency with the given information. However, there are two problems with this latter approach. First, it fails to give a good account of systematic protocols such as that in Table 1. And second, it merely shifts the burden of explanation to the operation of the consistency checker. How does the checking process know that (12) is consistent with the problem information without using procedures such as those in Tables 2 and 3?

Our suspicion is that whatever plausibility mental models have for these puzzles is due to the fact that they echo the output of the natural-deduction process. Notice, in particular, that the mental model in (12) corresponds to the atomic sentences on the left-hand side of Figure 1 and the mental model in (13) to the right-hand side. The only difference in representation is that the natural-deduction model also includes sentences corresponding to the original statements of the characters and to intermediate inferences. I would claim that the presence of the intermediate sentences is important since the protocol subjects sometimes said things like them. However, the important advantage is that the natural-deduction process explains exactly where all of these items come from. We could, of course, be mistaken in our conjecture of how a mental-models model of these puzzles would go; so we leave it as a challenge to mental modelers: Produce an explicit account of reasoning on knight-knave problems that is (a) theoretically explicit, (b) empirically adequate, and (c) not merely a notational variant of the natural-deduction theory. We don't believe such a challenge can be met and claim that such difficulties are symptomatic of a general failure in the mental-models approach (Macnamara, 1986; Rips, 1986).

There may indeed be alternative theories that can account for the results presented here, but the natural-deduction theory has a headstart. It is consistent with the strategies adopted by the protocol subjects, predicts error rates and solution times in the current experiments, and dovetails with earlier research on other deduction problems.

Postscript

It might seem to you that the success of our natural-deduction model depends on mundane empirical considerations such as those discussed in the preceding sections. We thought so too until a few days ago when someone we met – call him A – convinced us otherwise. It turns out that there is a *proof* that this cognitive model is correct, that is, correct on purely logical grounds.[4] Here is what A said:

(14) If I am telling the truth, then the natural-deduction model is correct.

This sentence must itself be logically true. For suppose that its antecedent is true. Then A is telling the truth and what he says – namely (14) – is true as well. The consequent of (14) then follows by modus ponens; so under the assumption that A is telling the truth, the model is correct. In other words, what we have shown is that *if* A is telling the truth, then the model is correct, which is exactly sentence (14). So (14) is indeed logically true. Of course, we still have to demonstrate the logical truth of the natural-deduction model, but this last step is easy. Since (14) is true and A said it, A must be telling the truth. Hence, the antecedent of (14) is true also, since that is what it asserts. By a second application of modus ponens, the natural-deduction model is correct. Q.E.D.?

References

Barwise, J., & Etchemendy, J. (1987). *The liar: An essay in truth and circularity*. New York: Oxford University Press.

Beth, E.W. (1955). Semantic entailment and formal derivability. *Mededelingen van de Koninklijke Nederlandse Akademie van Wetenschappen*, *18*, 309–342.

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., Reiser, B.J., & Rumain, B. (1984). Some empirical justification for a theory of natural propositional logic. In G.H. Bower (Ed.), *The psychology of learning and motivation* (Vol. 18, pp. 313-371). New York: Academic Press.

Cheng, P.W., & Holyoak, K.J. (1985). Pragmatic reasoning schemas. *Cognitive Psychology*, *17*, 391–416.

Clark, H.H., & Chase, W.G. (1972). On the process of comparing sentences against pictures. *Cognitive Psychology*, *3*, 472–517.

Clocksin, W.F., & Mellish, C.S. (1981). *Programming in PROLOG*. Berlin: Springer-Verlag.

Doyle, J. (1980). *A model for deliberation, action, and introspection* (AI TR-581). Cambridge, MA: MIT Artificial Intelligence Laboratory.

Evans, J.St.B.T. (1982). *The psychology of deductive reasoning*. London: Routledge & Kegan Paul.

[4] Perhaps A had been reading about Löb's paradox (see Barwise & Etchemendy, 1987, p. 23).

Føllesdal, D., & Hilpinen, R. (1971). Deontic logic: An introduction. In R. Hilpinen (Ed.), *Deontic logic: Introductory and systematic readings* (pp. 1–35). Dordrecht: Reidel.

Gentzen, G. (1969). Investigations into logical deduction. In M.E. Szabo (Ed.), *The collected papers of Gerhard Gentzen* (pp. 68–131). (Originally published as Gentzen, G. (1935). Untersuchungen uber das logische Schliessen. *Mathematische Zeitschrift*, *39*, 176–210, 405–431.)

Griggs, R.A., & Cox, J.R. (1982). The elusive thematic-materials effect in Wason's selection task. *British Journal of Psychology*, *73*, 407–420.

Holland, J., Holyoak, K.J., Nisbett, R.E., & Thagard, P. (1986). *Induction: Processes of learning, inference, and discovery*. Cambridge, MA: MIT Press.

Jaskowski, S. (1934). On the rules of supposition in formal logic. *Studia Logica*, *1*, 5–32.

Johnson-Laird, P.N. (1983). *Mental models*. Cambridge, MA: Harvard University Press.

Kripke, S. (1975). Outline of a theory of truth. *Journal of Philosophy*, *72*, 690–716.

Lewis, D. (1973). Causation. *Journal of Philosophy*, *70*, 556–567.

Lewis, D. (1974). Semantic analyses for dyadic deontic logic. In S. Stenlund (Ed.), *Logical theory and semantic analysis* (pp. 1–22). Dordrecht: Reidel.

Macnamara, J. (1986). *A border dispute: The place of logic in psychology*. Cambridge, MA: MIT Press.

McCawley, J.D. (1981). *Everything that linguists have always wanted to know about logic but were ashamed to ask*. Chicago: University of Chicago Press.

Osherson, D.N. (1974–1976). *Logical abilities in children* (*Vol. 2–4*). Hillsdale, NJ: Erlbaum.

Pollard, P. (1982). Human reasoning: Some possible effects of availability. *Cognition*, *12*, 65–96.

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

Rips, L.J. (1984). Reasoning as a central intellective ability. In R.J. Sternberg (Ed.), *Advances in the study of human intelligence* (Vol. 2, pp. 105–147). Hillsdale, NJ: Erlbaum.

Rips, L.J. (1986). Mental muddles. In M. Brand & R.M. Harnish (Eds.), *The representation of knowledge and belief* (pp. 258–286). Tucson: University of Arizona Press.

Rips, L.J. (1988). Deduction. In R.J. Sternberg & E.E. Smith (Eds.), *The psychology of human thought*. Cambridge: Cambridge University Press.

Rips, L.J., & Conrad, F.G. (1983). Individual differences in deduction? *Cognition and Brain Theory*, *6*, 259–285.

Smullyan, R.M. (1978). *What is the name of this book? The riddle of Dracula and other logical puzzles*. Englewood Cliffs, NJ: Prentice-Hall.

Stallman, R.M., & Sussman, G.J. (1977). Forward reasoning and dependency-directed backtracking in a system for computer-aided circuit design. *Artificial Intelligence*, *9*, 135–196.

Tarski, A. (1944). The semantic conception of truth. *Philosophy and Phenomenological Research*, *4*, 341–375.

Wason, P.C., & Green, D.W. (1984). Reasoning and mental representation. *Quarterly Journal of Experimental Psychology*, *36A*, 597–610.

Wason, P.C., & Johnson-Laird, P.N. (1972). *The psychology of reasoning*. Cambridge, MA: Harvard University Press.

Résumé

Les casse-têtes du type knights-knaves (véridiques-menteurs) se situent dans un univers ou certaines personnes, les *knights* ne profèrent que des vérités, tandis que les autres, les *knaves* ne profèrent que des mensonges. Par exemple, supposons qu'une personne A dise, "Je suis un knight et B est un knight", et la personne B dise "A est un knave". A est-il un knight ou un knave? B est-il un knight ou un knave?

Dans une étude pilote, nous avons demandé à des sujets de penser à voix haute pendant qu'ils étaient en train de résoudre de tels problèmes. Les résultats suggèrent que les sujets font des hypothèses à propos du statut knight/knave des personnages puis mènent des inférences déductives à partir de ces hypothèses pour en

tester la consistance. Ceci nous a encouragé à modéliser le processus avec une simulation fondée sur une théorie de la déduction naturelle formulée précédemment. Le modèle contient un ensemble de règles de déductions sous forme de productions et une mémoire de travail qui garde la preuve de la réponse correcte. Plus le nombre de pas (hypothèses et inférences) dans la preuve est grand, plus la difficulté prédite du puzzle est grande. Les expériences présentées ici confirment cette prédiction en montrant que les sujets font plus d'erreurs (Expérience 1) et mettent plus de temps pour résoudre (Expérience 2) des puzzles dont la preuve a un grand nombre de pas.