Stanford Law Review

November 1970

VOLUME 23   NUMBER 1

Stanford Law Review

November 1970

CONTENTS

Some Speculation About Artificial Intelligence and Legal Reasoning*

Bruce G. Buchanan†
Thomas E. Headrick‡

Although the computer has worked its way out of the laboratory and into common experience, lawyers have made slim progress towards finding useful computer applications. Research in artificial intelligence, a branch of computer science, has illuminated our capacity to use computers to model human thought processes. This research suggests that computer science may assist lawyers in both the study and performance of their reasoning processes. In this Article we will argue that the time has come for serious interdisciplinary work between lawyers and computer scientists to explore the computer's potential in law.

Interdisciplinary work between the lawyer and the computer scientist has floundered on the misconceptions that each has of the other's discipline. As a result, no one has yet attempted computer programs incorporating complex techniques of legal reasoning. Even efforts in legal information retrieval have been hampered by these misconceptions. In retrieval, lawyers have viewed the computer as, at most, a storehouse from which cases and statutes might be retrieved by skillfully designed indexing systems. Computer scientists, on the other hand, have operated with a conception of law as a collection of facts and "correct" legal principles; they have assumed that the computer can be most helpful to the lawyer if it can retrieve the right answers quickly. But the lawyer rarely looks for, or even expects, clear answers. More often than not, he searches his data base—treatises, articles, statutes, cases, and other materials—in order to construct legally acceptable arguments in the pursuit of one or more objectives. Likewise, the computers have information processing capabilities that are overlooked in the common conceptualization of computers as information storage devices. Work in artificial intelligence has been proceeding for more than a decade on efforts to model information processing after human thought

patterns. So far, the efforts in legal retrieval have given little consideration to the possibility that computers might operate on the legal data base the way a lawyer does. Yet the work in both fields—law and computer science—suggests that the computer modeling of legal reasoning would be a fruitful area for research. In this Article we speculate about the dimensions and possible directions of this research. Under the most promising of outcomes, interdisciplinary research could lead both to a greater understanding of the legal reasoning process and to the design of machine methods for performing parts of it.

The prospect of using computers to model legal reasoning processes is likely to prompt a typically lawyer-like response: So what if we understand legal reasoning or legal argument formation better? The effort to create programs to simulate legal reasoning processes should have at least two benefits: It should prompt more systematic study of legal problem-solving, and it should advance knowledge of the problem-solving capabilities of computers. Knowing more about the ways in which lawyers search and manipulate the legal data base might lead to improving the lawyer's skill at his work. We recognize the possibility that the work of many lawyers might actually involve little use of the legal data base for argument construction or dispute resolution. Until our knowledge about how lawyers really operate is firmer, however, we believe our expectation is plausible and acceptable.

I. ROUTINE LEGAL RESEARCH AND THE COMPUTER

It is useful to start with some conception of a legal research system to give us some notion of where the current legal retrieval systems might fit in a more comprehensive legal reasoning system and how current artificial intelligence research might contribute to the design of a system capable of performing both the routine and some creative aspects of legal research.[1]

A legal research program might perform many routine duties. At minimum, the computer should be able to retrieve statutes, case language, case summaries, and other legal data, to answer questions about that information, and to communicate in a natural (not cryptic) manner with lawyers. Automated information storage and retrieval systems have been developed that can store millions of records and search them more thoroughly and rapidly than a person can. Money is one major limiting factor in developing this computer skill. Not only is it expensive to gather, codify,

---

    1. For a general discussion of the problems of designing intelligent systems see Churchman & Buchanan, *On the Design of Inductive Systems: Some Philosophical Problems*, 20 BRIT. J. PHIL. SCI. 311 (1969). For a discussion of some of the benefits and problems arising from using computers in law, especially in predicting judicial decisions, see Stone, *Man and Machine in the Search for Justice*, 16 STAN. L. REV. 515 (1964).

and store large amounts of information, it is also expensive to run the computer programs that access all the information.

A few legal retrieval systems have been or are being developed.[2] These systems have two basic components: a data base and an indexing system. Usually the data base is the full text of statutes or cases, that is, the primary legal sources. The ingenuity in designing such a system lies in formulating the index, which provides the means of retrieving the legal material. Several methods have been designed. In the University of Pittsburgh-Aspen System, the only one commercially operational, the index is produced by the computer, which sorts and counts word frequencies in the legal sources.[3] For example, state statutes are read into computer storage. A frequency count is then made of all but the most commonly used words to compile an index of the words in storage. The searcher accesses the data base by requesting all statutory provisions containing the one or more words that he designates.

Slightly more complex are the association-factor retrieval systems that attempt to build search systems on the joint appearance and proximity of words in the same document. Words that appear together in the same document beyond some established level of statistical significance are assumed to be logically related to each other. Thus a user of such a system can retrieve all references to a given word and to words related to it at some specified level. For example, words related to "automobile" at a .75 level (or higher) in a particular document may include "car," "truck," "transportation," and "highway," in which case the user interested in automobiles can see these related references as well. Two experimental projects have based their indices upon this assumption.

One system of this type, developed at George Washington University, makes a set of comparisons and calculations to produce an index that associates any one term with a number of related terms.[4] The specification of a term by the user first produces a list of related terms. The computer then locates documents with the main terms and the related terms and,

---

2. In our discussion of retrieval systems we are indebted to a review of past and contemporary efforts prepared by Clifford Weaver, who was a Stanford Law and Computer Fellow in the summer of 1968. For other recent surveys see Comment, *Automated Legal Information Retrieval*, 5 Houston L. Rev. 691 (1968); and Note, *Science—Computers—The Use of Data Processing in Legal Research*, 65 Mich. L. Rev. 987 (1967).

3. The University of Pittsburgh-Aspen System is described in several places, including: Horty, *The "Key Words in Combination" Approach*, 1962 Modern Uses of Logic in Law 54; Horty, *Use of the Computer in Statutory Research and the Legislative Process*, in A.B.A., Computers & the Law 48 (1966); Kehl, Horty, Bacon & Mitchell, *An Information Retrieval Language for Legal Studies*, 4 Com. of the Ass'n for Computing Mach. 380 (1961); and promotional materials obtainable from Aspen Systems, Webster Hall, Pittsburgh, Pennsylvania.

4. Stiles, *The Association Factor in Information Retrieval*, 8 J. Ass'n for Computing Mach. 271 (1961); Eldridge & Dennis, *The Computer as a Tool for Legal Research*, 28 Law & Contemp. Prob. 78, 92 (1963). For a description of later work at George Washington University see Kayton, *Retrieving Case Law by Computer: Fact, Fiction and Future*, 35 Geo. Wash. L. Rev. 1 (1966).

by using previously determined computations of the relevance ratios between terms, assigns a relevance value to each document produced.

A somewhat similar system is under development by the American Bar Foundation in cooperation with I.B.M.[5] In this system the computer analyzes the frequency of co-occurrence. Words that co-occur frequently and with somewhat consistent intervening word space are assumed to be logically related. From this basic computation the computer constructs a grid with each word assigned a point. The expectation is that a section of the grid would contain words closely linked with each other, and that the machine would thereby have developed a thesaurus. That is, each word is stored in the thesaurus close to the other words of the text that are logically related to it (and to each other). The user initiates a document search, then, by naming one of the terms in the thesaurus. The computer searches for that word and others (or combinations of others) close to it in the grid to produce the references that mention any of these related terms.

These processes aid a researcher in finding all the documents that might have some conceivable relationship to the problem under search. They are machine-produced aids to his imagination. The attempts at computing document relevance help the researcher to design a strategy for reading the output. These systems, however, do not aid the researcher in the more important task of processing the output to achieve his desired objective.

Whether simple or complex, all of the legal retrieval systems proceed on common assumptions about the strategy of the legal researcher. First, they assume that he can be led to relevant material through the identification of dominant single concepts or generalizations. In other words, he can pick a set of key words that are relevant to his research objective, and these will lead him to the cases or statutes he needs. Second, they assume the researcher proceeds by identifying all the statutes and cases he can find that mention the designated concept. They tend to leave out other filters that the lawyer may use. Some are obvious, like age of a case, jurisdiction, the procedural context, and key facts. Others are less so, like the way a court characterizes an issue, matters to which judicial notice is given, cases followed or distinguished, and many others. Third, they all assume that the computer can do nothing more than fetch the material from its memory and that the researcher must digest what is produced in order to construct his argument or advice. All of these assumptions are questionable. They ignore both the objectives of legal research and the cognitive potential of computers. Lawyers can use more than routine help, and computers may be able to provide it.

---

5. Dennis, *Status of American Bar Foundation Research on Automatic Indexing-Searching Computer*, 1965 MODERN USES OF LOGIC IN THE LAW 131; Eldridge, *The American Bar Foundation Project*, 1965 MODERN USES OF LOGIC IN THE LAW 129; Eldridge & Dennis, *supra* note 4.

A more sophisticated program would be able to "understand" and process some (but not necessarily all) of the material about the items in its files. It could read stored items to find the best ones for the lawyer instead of merely pulling out the items that use any or all of the key words he has specified. It could know that Judge Banks did not necessarily hear only cases involving banking institutions, for example, or that a case about the misdeeds of a parent corporation was not about child abuse. Some understanding of the meaning of sentences, that is, some knowledge of semantics, is necessary to avoid such blunders and to provide more help to the lawyer.

This semantic capability is a more difficult task for computers than is key-word searching. Although there has been substantial computer research in the area of natural language communication, the present systems are limited by the difficulty of giving a computer program enough semantic processing power to understand English sentences. The programs that have been developed all work with restrictions either on the domain of discourse or on the number of facts and concepts that can be related. For example, one program works on algebra word problems in English;[6] another works with English sentences, mostly upon part-whole relations between objects;[7] others have attempted to relate facts, expressed in English, about any subject or set of concepts,[8] but these can so far handle only a limited number of facts and concepts. In restricted subsets of English, some existing programs can carry on respectable dialogs with humans. For example, a program has been written for performing the groundwork of a psychiatric interview.[9] Another program can conduct tutorial sessions with students on any restricted topic about which it has been instructed —for example, the special theory of relativity or the interpretation of a particular line of poetry.[10] Many other examples are cited in a recent survey article.[11] Thus, for the more routine language skills that the research system demands, computer capability has been demonstrated, at least for restricted problems.

The practical problems of applying these pioneering programs to as

6. Bobrow, *Natural Language Input for a Computer Problem-Solving System*, in SEMANTIC INFORMATION PROCESSING 135 (M. Minsky ed. 1968).

7. Raphael, *SIR: A Computer Program for Semantic Information Retrieval*, in SEMANTIC INFORMATION PROCESSING 135 (M. Minsky ed. 1968).

8. Quillian, *Semantic Memory*, in SEMANTIC INFORMATION PROCESSING 216 (M. Minsky ed. 1968); Black, *A Deductive Question-Answering System*, in SEMANTIC INFORMATION PROCESSING 354 (M. Minsky ed. 1968).

9. Colby & Enea, *Heuristic Methods for Computer Understanding of Natural Language in Context Restricted On-Line Dialogues*, 1 MATH. BIOSCI. 1 (1967).

10. Taylor, Automated Tutoring and Its Discontents, in ELIZA: A Skimmable Report on the ELIZA Conversational Tutoring System (E. Taylor ed. March 1968) (MIT Education Research Center, Cambridge, Mass.); Weizenbaum, *Contextual Understanding by Computers*, 10 COM. OF THE ASS'N FOR COMPUTING MACH. 474 (1967).

11. Simmons, *Natural Language Question-Answering Systems: 1969*, 13 COM. OF THE ASS'N FOR COMPUTING MACH. 15 (1970).

large and complex an area as legal research are formidable indeed. Moreover, because the range of topics, situations, and concepts covered by law is virtually unbounded, a formal encoding of cases and laws probably could not preserve the richness, the suggestiveness, or the fertile ambiguities of the English language. Yet computer science is making progress with natural language communication, and its long-range relevance to legal problem-solving is clear.

These problems are still not in the same class as the problems of raising the computer from a quick librarian to a sophisticated information processor. More difficult problems lie ahead for lawyers, or anyone else hoping to let computers participate in some of their tasks. While it is relatively easy to say what routine skills would be desirable in the program and to point to relevant computer research, it is not so easy to say what a lawyer does as a creative researcher (at least not in exact terms), and it is even harder to conceive of the computer's potential. Even so, the current research in artificial intelligence suggests that both problems deserve further inquiry.

II. GENERAL PROBLEMS OF PROGRAM DEVELOPMENT

Understanding and codifying the decisionmaking processes of lawyers presents one of the greatest challenges of the proposed research. Apart from the computer program that could emerge, one of the payoffs of this research is the insight that lawyers will gain into their own modes of thinking. Such a payoff has been realized in other fields where computer scientists prodded specialists for their rules, which were then put into the developing program.[12] Lawyers, like many specialists, can be expected to respond to questions about their methods, their rules of thumb, and their decision rules with doubts about the propriety of the questions. In the absence of any reason to speculate on how they carry on their work, they now apply complex sets of rules without being much aware of the rules themselves. But demands for an intelligent computer system to aid lawyers cannot be met unless lawyers themselves are able to articulate their methods and thought processes.

Computer scientists working in these areas share two assumptions about human problem-solving: (1) problems can be broken down into a set of subproblems, and (2) the solution to any subproblem requires a series of decisions that are governed by decision rules. As mentioned earlier, humans who solve complex problems often cannot articulate the principles by which they simplify their problems or the decision rules by which

---

12. For example, organic chemists have come to understand their own analytical rules better as a result of artificial intelligence research constructing a computer program to perform analyses of some experimental chemical data. *See* notes 19–20 *infra*.

they solve the subproblems. It would not be necessary for a lawyer to be able to articulate fully his procedures before computer research in legal problem-solving can begin. It would be necessary, however, for him to believe that the methods of solution can be made precise.

Many problems are likely to be encountered in searching for programmable rules used by lawyers. Experience in artificial intelligence research has shown that this is a formidable barrier to the creation of any intelligent computer system. Searching for statements of rules used by an individual in solving problems or constructing arguments is bound to be uncomfortable or unsettling. It is hard work, and it often causes the subject of the inquiry to rethink his fundamental reasons for holding one set of principles rather than any other. It may even point out the lack of substantial rational grounds for many decision rules.

Even if an individual lawyer is willing to attempt this exercise, another set of problems stands in the way of creating the computer program. Lawyers who are unaccustomed to the rigorous demands of computer programming languages will feel frustrated with the gap between what they want to say and what the computer language lets them say. This frustration is common among programmers as well. Its cause does not lie in the restrictions on the content of the rules that are to be programmed. Instead, the cause of the frustration is usually the disparity between the natural statement of the rule and the formal statement within the programming language. The problem lies in representing the rules so that the computer can use them.

There is another important problem that is well known to those in the field of artificial intelligence: Even after one machine-readable representation has been found for a class of problems, one has no assurance that it provides the computer with the best method of handling all problems of that class. Many times success or failure in solving a complex problem will hinge on representing the problem correctly at the start.[13] So the search for rules used by a lawyer will not be over as soon as he can make some statements about his priorities for legal principles and his heuristic rules.[14] He must also be prepared to recast these statements into

---

13. For example, the so-called "mutilated checkerboard" problem is easy in one representation, but difficult in another. The puzzle is this: When trying to cover a checkerboard with dominoes, each of which covers two squares, it is obvious that removing one corner square from the board renders the solution impossible. Can it be completely covered, however, when two diagonally opposite corners are removed? The statement of the problem suggests that actually laying dominoes on a checkerboard might be the way to find out. However, this is the wrong representation. The one that makes the solution trivial emphasizes the red-black coloring of the board and the fact that each domino must cover both a red and a black square. Now it becomes obvious that a checkerboard with two red squares removed cannot be covered with dominoes. The solution, given the right representation, is equally trivial for human or machine problem-solvers.

14. Heuristics (or heuristic rules) are rules of thumb that usually contribute to solutions of problems without in any way providing a guaranteed method for solution. In a chess game, for example, one heuristic for winning is to control the center of the board. There is certainly no guarantee that the player controlling the center will win; yet center control has contributed to winning enough games that it is a guideline for every player.

other representations. Perhaps a more formal representation will facilitate programming; perhaps using more general concepts will increase the range of legal problems that can be solved; or perhaps using more specific concepts in some places will make some problems almost trivial.[15] At any rate, this is an important and difficult consideration for the designer of a system that does legal reasoning.

III. Two Models of Creative Legal Research

We know too little about the styles and structures of legal research strategies. Although lawyers do legal research and solve legal problems day in and day out, systematic analysis of this process has been rare.[16] As a consequence, our models of the legal research process are incomplete and oversimplified. We have no solid base of data on the legal research process, and, of course, have not attempted an exhaustive description of legal thought processes. The two models discussed here have been derived from introspection and tested upon our colleagues. They provide a framework for looking at lawyers' underlying thought processes by describing a portion of their cognitive tasks.

In the first model, the factual situation has been set, certain events have happened, and a client wants to know what his rights, remedies, and risks are. In the second, a client is looking for guidance for a future action, and the client can control the facts or events somewhat to suit his legal needs as assessed by the lawyer. For both models we have assumed that problem identification has taken place: The lawyer has already delimited his area of concern by concluding that he has a tax problem, an agency problem, a contract problem, or some other. He is ready to concentrate on one area of law. How he arrived at that point involves a complex search of the space of laws and legal problems. The existing and proposed legal retrieval systems model a part of this search process, but, as we indicated, at present they leave out other parts.

A. *Model I: Interpreting the Facts and Law to a Client's Benefit*

Suppose a client has been involved in an automobile accident or he has in some way broken off a contractual relationship, and someone is threatening to sue him. The events have already occurred. First of all, the lawyer would rarely be looking for a direct answer that his client is liable or not liable. Rather, he would seek to construct a legal argument that would justify the actions of his client. The lawyer would attempt to generalize

---

15. For example, mentioning red and black squares and not just adjacent ones in the mutilated checkerboard problem described in note 13, *supra*, makes the solution obvious.

16. Two useful accounts, both clearly based on introspective study and case analyses rather than empirical investigation, are C. Morris, How Lawyers Think (1938) and E. Levi, An Introduction to Legal Reasoning (1949).

the already-occurred factual situation and select the legally relevant facts. He does this by reference to legal rules or concepts that make some facts relevant and others irrelevant. He looks at the facts and checks them against potential linkages of rules that would lead to the result he favors and avoid the result he disfavors. Although he may start with one conception of the facts, he may end up emphasizing other facts because of the way in which he constructs a legal argument (that is, a linkage of concepts to the desired legal result).

The skillful advocate strings together rules in a way that justifies the result he is seeking and at the same time encompasses the factual occurrence in a way that makes the rules he has selected appear to be the ones best applicable to the situation. His research strategy is to identify the string of rules that both leads to a desired result and plausibly encompasses a set of facts that accounts for what has happened to his client.

This model may be clarified for the reader without legal training if we spell out more concretely some of the lawyer's possible approaches to the problems presented by the factual situation in, for instance, a simple contracts case. In *Boston Ice Co. v. Potter*,[17] an ice company sued to obtain the purchase price of ice it had delivered to a customer. Two years earlier that customer had switched its business from Boston Ice to a rival company. Boston Ice had then purchased the rival company and resumed delivery to the customer without informing him of the change. The customer refused to pay when he finally learned the identity of his supplier.

The goal of a lawyer representing Boston Ice would be to secure a judgment against the customer for the value of the ice delivered. By one line of argument, he might try to establish that the plaintiff and defendant had contracted with each other. The facts show no direct negotiations between the plaintiff and defendant leading to an express contract, so he would have to argue that a contract should be implied from the situation. Perhaps our lawyer would reason that delivery of goods was an offer and their use an acceptance, and then search for cases with that holding. If he turned up some cases fitting this generalization, he would then want to know whether anything in the nature of either the delivery or the goods distinguished those cases from his client's. For instance, assume he found a case holding that a water company's delivery of water constituted an offer and its use an acceptance even though a new resident in a house had never communicated with the water company. That case might be distinguished on a variety of grounds: the presence of an established network of pipes in which the water company had invested; the need to deliver water by a single closed system in order to assure water quality; the existence of a

---

17. 123 Mass. 28, 25 Am. R. 9 (1877).

state law requiring all dwellings to have running water; or because the water company's monopoly gave the new resident no choice of alternative supply. Our lawyer would have to face one uncomfortable fact in his client's situation: The defendant had previously broken off his dealings with the plaintiff and turned to a competing ice house. If he had known who was supplying him ice, he would likely have terminated the deliveries. Our lawyer might fear that cases supporting the principle of implied offer and acceptance are insufficient because they cover factual situations that do not include this element. He would then look for a less factually restrictive line of cases.

He may find some cases that suggest that an assignment of a contract is valid under some conditions. This validity, he will learn as he reads a few cases, applies to contracts that involve fungible goods or that do not involve some element of personal service in their performance. Ice would appear to meet the first condition, but this case may well come within the personal services exception. Was the defendant receiving ice alone, or ice and some element of personal services? Did he previously terminate his arrangement with the plaintiff because he found the service unsatisfactory? The questions suggest that our lawyer would need to re-interview his client, and possibly take the defendant's deposition, to determine if he could pin down facts that would place his client within this line of cases.

Our lawyer might pursue other avenues, but this account is complete enough to illustrate elements of the argument construction process stressed in Model I: specification of an objective; an initial statement of facts; search for cases offering a rule that leads to the desired legal consequences on the basis of those facts; evaluation and comparison of the facts and legal issues in the different cases; testing of the given facts against a rule; acceptance, rejection, or modification of the rule, or the resulting search for additional facts. For these processes the lawyer does not need all possibly relevant cases; he needs only a solid sample that indicates how previous courts have usually reasoned and ruled in analogous situations.

B. *Model II: Recommending Actions that Satisfy a Client's Goals and Avoid Unfavorable Consequences*

In the second model the lawyer is no longer involved merely in the argument formation process, but also is concerned with risk assessment with respect to future actions. He starts by identifying the typical situation where the law would sanction the client's proposed actions. Then he checks the law and the potential facts of his client's situation to see if there are any reasons why these desired typical solutions are inapplicable. If he finds some reason, he would then attempt to restructure the existing facts, the legal

generalizations of them, and the proposed action to develop a better fit. He would continue this restructuring and testing until he finds the formulation of existing facts, proposed actions, and legal consequences of them that minimizes the risks and maximizes the sought objectives. Model II differs from Model I in that the facts become variable to a greater extent and the objectives are not simply a favorable legal result, but also combinations of legal, business, and other goals. The Boston Ice situation will again illustrate the lawyer's method.

Suppose the ice company, upon buying a rival company, asks our lawyer what risks it runs if it simply continues supplying the other company's customers. He has some control over the facts since he can suggest actions his client might take. These actions will improve the client's chances of fitting his situation within a legal rule leading to favorable legal consequences: in this case, prompt payment of bills and minimization of the need for litigation to collect bad debts. For instance, the safest legal position for his client would be to make express contracts with the new customers. He might, therefore, issue express offers to them and solicit clear acceptances before beginning deliveries. But this course entails some risks. Some proportion of the customers may decline the offer and turn elsewhere for ice. On the other hand, if the client merely sends a letter announcing the purchase of its rival's business, he encounters the risk that a court may rule that to be insufficient to create an obligation from the new customer. Our lawyer might make other suggestions, each entailing some measure of legal risk, which must be balanced against business risks. The trade-offs may not be clear. The lawyer and the client must combine judgment as well as they can.

How does the lawyer proceed? Working within the general conceptual framework of contracts, he can think of a number of actions his client might take. From his legal data base he can extract commentary and cases that help him predict how a court is likely to react to cases involving each of the possible sets of facts. Some sets of facts will appear safer for the client than others; the lawyer will go through the process of matching facts and generalizations about facts in the cases and treatises with the imagined fact-sets involving the client and potential customer litigants to determine whether the fit is good enough to dissuade the customers from risking a lawsuit. His assessments of the varying strengths will enable him to make a rough ranking of the possible actions in order of their legal risk. Although the overall process differs from the first model, with its unpaid bill and pending litigation, many of the subprocesses involving manipulation of the legal data—facts, rules, and their consequences—are similar.

IV. THE THOUGHT PROCESSES OF A LAWYER

The existing retrieval systems help only on the periphery of the processes described in both models. They retrieve cases and statutes that are potentially relevant to some of the facts under consideration. Ideally the lawyer would not want from the computer mere lists of statutory provisions and cases related to legal concepts that may, after he wades through them all, help him put together an argument or design a new search. He would want a system that would produce legal arguments—the end products of his research.

To begin designing such a system, we have to know more about the mental processes a lawyer uses to solve his legal problems. Only in this way can we begin to structure the processes so that a computer could imitate them. We can identify some of the lawyer's mental steps from our models. Certainly there are many others.

First, the lawyer establishes and pursues a goal. He seeks some satisfactory legal result for his client. There are some crucial steps in this process. One is his perception of linkages, of how a set of facts calls into play a rule, which then calls into play another rule or set of consequences either compatible or incompatible with his goal. Another is his decision about whether an indicated legal result is compatible or incompatible with his goal. One set of mental processes, therefore, concerns establishing goals, finding linkages from facts to rules to legal consequences, and measuring at various stages in the research process the compatibility of a set of consequences with the established goal.

In the Boston Ice litigation example, the lawyer had a clear objective: payment to his client for the ice. He tested, in succession, the applicability of several possible legal characterizations of the facts—express contract, implied contract, valid assignment. Each involved a linkage from facts through a rule to legal consequences. He accepted or rejected a specific linkage on the basis of how well the facts of his client's case fit with other cases that had applied the rule and resulted in certain consequences. In that process, he inevitably read cases that, though based on similar facts, applied rules that lead to undesirable consequences. He sought the rejection of those rules by distinguishing the facts.

Second, there is the process of fact recognition and characterization. The facts suggest some possibly applicable rules; the rules and the cases using them suggest the relevance and importance of certain facts. The rules that are being explored will influence the decision about which facts are relevant. Working with these interrelations is one aspect of recognizing relevant facts in legal problem-solving.

A related aspect of legal research concerns the use of other cases and their rules as bases for arguments in a client's case. There is no uniform way in which a judge or a lawyer records the facts of a case. Facts may be events like the collision of A's car and B's car; they also may be relationships like C was a passenger in A's car. Moreover, what passes for a fact in one case may become a question of legal dispute in another. For instance, in the *Boston Ice* case the prior termination of a contract was treated as a fact; in another it may be the crucial point at issue. In addition to the usual ambiguities of language, one often finds legal characterizations of events and relationships treated as facts for the case at hand. Two cases might have involved identical events, but in one the court described events, and in the other another court summarized them with a legal conclusion. The perceptive lawyer will recognize the similarities and differences between the facts of the two cases.

A third process that a lawyer performs is rule selection, that is, identifying which one of several rules applies to a factual situation. One simplified model of legal reasoning portrays a collection of facts classified by legal rules from which specified legal consequences flow. In conception, the legal rule is an if-then statement: If facts $A_1 \ldots A_N$, then legal consequences $B_1 \ldots B_N$. In practice, however, the rule that may apply rarely emerges so neatly. In Boston Ice, the account of the past behavior of the parties could be used to support the application of two different rules leading to opposite legal conclusions; that is, the absence of express contract or the presence of a valid assignment of a contract. The legal problem-solver often differentiates in some way between two rules that might apply to the same behavior and events. He thus has some way of resolving rule conflict. The form of differentiation chosen has to do with his objectives, with the relative weight he attaches to the various facts, and with the inferences he draws from some facts on the basis of his knowledge of additional facts. In effect, he finds or constructs another rule to resolve the rule conflict.

A fourth process concerns analogies. A lawyer usually prefers to fashion arguments built from cases whose facts are similar to the facts with which he is working. If he cannot find such cases, he resorts to finding cases with facts that are analogous to his own in formulating an argument. One method of finding an analogy is through generalization of a legal rule. Although in one case a rule was applied to a specific set of facts, the language in which the court stated the rule may deductively allow its application to a different set of facts. The rule may use words that are capable of encompassing a variety of events, actions, or relationships. In our Boston Ice example, for instance, the lawyer examined whether the case of water

delivery would hold for ice deliveries. In addition, analogies are found by generalizing on the factual situations.[18]

V. Current Computer Work and Legal Thought Processes

The four legal problem-solving processes we have isolated are the following: (1) finding conceptual linkages in pursuing goals, (2) recognizing facts, (3) resolving rule conflicts, and (4) finding analogies. Some computer science research in the areas of artificial intelligence and simulation of cognitive processes bears directly on these creative aspects of legal research. We will review some of these developments briefly and draw attention to possible applications to legal research. One particular program, named "Heuristic DENDRAL,"[19] will be discussed in some detail to illustrate the present capability of a current program. By this illustration, we hope to convey the present power and limitations of artificial intelligence methods.

The task of the Heuristic DENDRAL program is to make inductive inferences from experimental data to explanatory hypotheses in a complex area of science. The data are analytic data resulting from fragmentation of organic chemical molecules within a mass spectrometer. The explanations the program seeks are models of the original chemical compound that produced the data. Using general principles, the program tries to relate the facts (data) to an hypothesis that clarifies the facts or shows why the facts should be expected. It attempts to reason from the facts to the most plausible conclusion, given general principles governing the relation of facts to conclusions.[20]

18. For example, assume a case holding that a corporate officer who had access to information that should have forewarned him of the fraud of a potential creditor was liable to his stockholders. From this case it could be argued, by analogy, that the airline officer who had access to information about impossible flying conditions but authorized a flight that took people to their death was also liable for negligence. The factual situations would be made analogous in this instance by isolating the notion of access to information that should have forewarned the defendant.

19. The Heuristic DENDRAL program was conceived by Professors Joshua Lederberg and Edward A. Feigenbaum at Stanford University after C. West Churchman of the University of California at Berkeley suggested studying the design of inductive systems. The program was mostly written by Georgia Sutherland, Allan Delfino, and Bruce Buchanan. It currently runs on the IBM 360/67 computer at Stanford. See Buchanan, Sutherland, & Feigenbaum, *Rediscovering Some Problems of Artificial Intelligence in the Context of Organic Chemistry*, in 5 MACH. INTELLIGENCE 253 (B. Meltzer & D. Michie eds. 1970).

20. The Heuristic DENDRAL program has been discussed in detail elsewhere. *See id.* The program itself is conceptually simple and can be described briefly for those intersted.

The first phase of the program accepts the facts (*i.e.*, the analytical data) and attempts to make some sense out of them by deciding which facts are relevant to the problem and constructing a plan for its solution. This first step has been labeled the planning phase of the program for this reason. The data come from a mass spectrometer, an instrument that bombards the molecules of the sample with electrons and collects the resulting fragments that have become charged in the process. The mass spectrum is usually presented to the chemist as a bar graph with the x-points representing masses of the fragments that were charged, and thus recorded. Each y-point of the mass spectrum represents the relative abundance of fragments of the corresponding mass. The problem is more difficult than fitting together pieces of a jigsaw puzzle because not all the fragments produced are recorded—only the positively charged ones are. Also, not all the recorded points represent results of a simple frag-

As most treatises on induction specify, the general approach to this problem is through the steps of analyzing the facts, constructing plausible hypotheses, and evaluating (verifying, validating) the hypotheses. The initial problem of explaining the data is thus broken down into three subproblems. Each of these, in turn, is little more than a set of subproblems. At the lowest level, the subproblems are so straightforward as to be called decisions.

The success of Heuristic DENDRAL, as with almost all nonnumerical problem-solving programs, lies in its ability to define the totality of possible solutions and search the totality efficiently for the best one. An important part of this task is constructing a search plan carefully enough to reduce the size of the search space[21] and still include the correct solution in the space. Although there are many variations on this generate-and-test strategy, its fruitfulness for other artificial intelligence programs suggests that it may be a useful approach to legal applications.

As in chemistry, a legal problem has an almost countless number of possible solutions, some better than others. Heuristic DENDRAL explores

---

mentation of the molecule. Many secondary and tertiary processes occur after fragmentation but before recording. Moreover, as with any method of collecting data—including surveys and personal observation—it is virtually impossible to avoid errors of one kind or another or to record only the data that will turn out to be the most significant. Usually it is a mistake to try.

So the first thing the program does is to apply some "common sense chemical rules" to get rid of the most blatantly spurious pieces of data. The program attempts to mirror the chemist's first adjustment of the data in this first stage. It rejects data that are either obvious mistakes or that are irrelevant to the solution.

The next part of the planning phase scans the filtered mass spectrum looking for patterns that indicate one or another class of solutions. For example, the class of molecules known as ethers exhibit different characteristic patterns than the alcohols. This planning process is grounded in almost exactly the same rules of mass spectrometry as the chemist uses for these classes of compounds.

The plan that results from the first phase of the program isolates important features of the final solution, without filling in the details. Although many chemical molecules may be compatible with the plan, only molecules that contain the structural features indicated by the plan will be considered plausible explanations of the data. In other words, the plan indicates a general class, within which must lie the solution of the original problem. One such plan would indicate the class of ethers that have five carbon atoms on either side of the oxygen atom.

The second phase of the Heuristic DENDRAL program generates specific instances of molecules within the framework of the plan. Instead of relying on something as vague as "intuition" to produce specific hypotheses, the program has a systematic generator. The virtue of this structure generator is that the program can construct hypotheses in a way that guarantees that every instance of the general class will be generated, if needed, until the class is exhausted. It also guarantees that no hypothesis will be considered twice and no equivalent hypotheses will be generated. The result of the second phase is a list of models of molecules compatible with the plan. If the plan is very specific, the list is very short; in many instances the plan is so good that only a single model fits its constraints.

The last phase of the program tests the candidate solutions (the models of molecules) produced by the generator in an attempt to find the best explanation of the data. Some of the consequences of each hypothesis are predicted and tested against additional data—either parts of the mass spectrum ignored earlier or the results of an entirely different experiment, a nuclear magnetic resonance spectrum, for instance. If the prediction is inconsistent with the data, that hypothesis is rejected. The candidate explanations that remain are then ranked on the basis of how well they explain the original data. The conclusion of this whole process is the choice of the best solution (or the best set of solutions) to the original problem.

21. A search space is an organized, possibly infinite, set of possibilities to be considered for the solution of a problem. Chess playing programs, for example, decide which move is best by exploring several of the possible moves and responses to them in the total search space. When a search space is very large, as in chess, a computer program must decide which parts of it to explore in detail and which to ignore.

the space of possible explanations by generating larger and larger pieces of chemical molecules, checking for plausibility each time. If some similar method could be devised for exploring the space of legal arguments systematically, the problem of selecting the best solution(s) could be reduced to deciding whether the alternative paths were likely to lead to a reasonable solution. The plan constructed by the chemistry program points to a small part of the total space, namely, the part that seems most likely to contain the best solution. Possibly a planning program could look at the data of a legal problem to suggest constraints on the search space. One such constraint on the solution space that lawyers determine initially is the category of the problem, which is what we have referred to as problem identification. Heuristic DENDRAL is doing what a lawyer does when he identifies his problem in a generic way and constructs a list of plausible (or possibly applicable) rules, which then, by another set of thought processes, are tested by a closer look at those rules as applied in previous cases, and the facts in the client's case.

A. *Pursuing Goals*

The goal of the Heuristic DENDRAL program is to find an explanation of the given empirical data. As in many legal problems, this goal is only implicitly defined. The program works from the given data toward the final solution, using its knowledge of the properties of the solution. For example, the DENDRAL solution must be a model of an organic molecule that is consistent with the analytical data. The solution must also account for many (or most) of the prominent data points. The program tries to reach its implicitly defined goal by reasoning from the given data toward the goal. When it is obvious to the program that a path in the solution space does not lead to the goal (*i.e.*, does not lead to a molecular model that satisfies the criteria of solution), that path is abandoned. Often, several alternative paths must be traversed to the end, each terminating in a candidate solution. In such cases, post facto measures of acceptability are applied to determine which candidate is best. The candidate exhibiting the most properties of the implicitly defined goal is identified as the best solution to the problem.

In the "General Problem Solver," a widely known computer program written by Newell, Shaw, and Simon,[22] the goal is explicit at the start. The program is expected to find a line of reasoning that takes it from the premises of a problem to the stated goal. For example, given the set of axioms of truth-functional logic and a theorem to prove, the program

22. Newell & Simon, *GPS, A Program that Simulates Human Thought,* in COMPUTERS AND THOUGHT 279 (E. Feigenbaum & J. Feldman eds. 1963).

tries to find a series of rule applications that transform an axiom into the theorem. This program was designed to solve puzzles of the form "How do I get there from here using the means given?"[23]

In a similar way, the lawyer takes sets of facts and attempts to apply legal rules or concepts to those facts in order to achieve a favorable result for his client. Problems that fit Model I above also seem amenable to solution by the goal-seeking strategy of the General Problem Solver. Theoretically, at least, one can conceive of a legal problem-solver that would contain legal rules, ways of testing their application to given fact situations, and a way of measuring progress toward the goal. After someone prescribed a set of facts and the desired results, the problem-solver would compare those facts (or subsets of them) to the situations under which the stored rules apply, apply the appropriate rule, and compare the legal consequences with the lawyer's goal. Gradually, then, it could build an argument leading to the desired result.

A computer program has been written in the context of investment portfolio selection that solves problems like those under Model II.[24] The task of this program is to select an investment portfolio that helps the client realize his investment goals. Here, as in the legal context, the problem is to advise actions the client can take to help him meet his goals. The program processes information on the economy, industries, and individual companies to produce a list of acceptable securities. Basically the program applies some general rules to filter out a set of possible investments; it then applies to this set some particular rules related to the interests of the investor to produce, finally, a balanced portfolio for the client. Conceivably, the same process might be applied to Model II legal problems, where the lawyer attempts to realize a client's objectives by legal routes that minimize adverse legal consequences.

Admittedly, it requires a substantial jump to convert Heuristic DENDRAL, the simulation of the investment trust officer, or the General Problem Solver into a program that can handle even some of the complexity of law. Yet even if, for the purposes of experiment, this complexity were artificially circumscribed, it might be useful to compare some variations of one of the goal-seeking programs operating on a legal problem with the processes a lawyer uses to solve the same problem. The comparison would probably lead to a refinement of our legal research models.

---

23. For example, given the starting locations of three cannibals and three missionaries on one side of a river and the goal of having all cannibals and missionaries alive and well on the other side of the river (no more than two persons are allowed in the boat at a time, and the cannibals must never outnumber the missionaries on either side of the river if the "alive and well" clause is to be satisfied), the program tries to find a series of boat trips that transforms the initial state into the goal state.

24. Clarkson, *A Model of the Trust Investment Process*, in COMPUTERS AND THOUGHT, *supra* note 22, at 347. The program was written with the purpose of modelling the thought processes of a particular investment trust officer in a medium-sized national bank.

B. *Recognizing Facts*

No existing program shows that computers can successfully recognize and reinterpret factual statements, the second of four legal problem-solving processes that a sophisticated legal program should have, but one is at least conceivable. The ability to find and characterize facts is an important but elusive element of human problem-solving, in scientific work as well as in law. Computer work has been started; much more remains to be done.

Fact recognition is the first step of the Heuristic DENDRAL program. Given the experimental data, the program, like any scientist, must distinguish the "real" data points from the spurious. It separates erroneous data and uninformative pieces of data from the facts that are important for solving the problem. The program can also add pieces of data that "should" have been in the original set based on inferences from theory and the data actually appearing. Here, and throughout the program, the decisions are based on the decision rules used by experts. They are usually grounded in theory, but often include heuristics.

With a good deal more theory (or more risk), the program could eliminate data points it decided were not useful for the problem even though they passed the general test of legitimacy. The rules for usefulness would be applied at this point in the program in much the same manner as the rules for spuriousness. Although the program would save some time at later stages because there would be fewer facts to analyze, the small savings and the higher risk of excluding data necessary to arrive at the best set of solutions both seem to make such filtering undesirable for this program. Thus the Heuristic DENDRAL program does not eliminate as many data points as it might while looking for the "real" facts relevant to a problem. But the mechanisms for discarding useless facts already exist and could theoretically be applied to reduce a data base to a manageable size.

Classifying facts and factual situations is also an important part of creative legal research. A concept-formation program written by Hunt and Hovland suggests a means of devising classification rules for a given set of objects.[25] This program models a psychological experiment in learning where an individual is presented with a series of objects described as A's and not A's, and he is asked to devise a rule explaining the classification. The program has three phases of operation. Ordering routines take note of common characteristics of the members of a class. Description routines identify common relationships among the characteristics each member of the class may have. Finally, solution routines try conjunctions and dis-

---

25. Hunt & Hovland, *Programming a Model of Human Concept Formation*, in Computers and Thought, *supra* note 22, at 310.

junctions of these common relationships as classifying rules. The concept-formation problem is thus broken into distinct subproblems.

Recognizing commonality and differences in factual situations and formulating concepts that link the commonalities and distinguish the differences is a substantial part of the process of legal argument formulation. This process occurs when a lawyer compares cases that deal with the same concept but have differing results. For instance, when research turns up a series of cases dealing with contributory negligence, it becomes important to weave an argument that links together those factual situations in which the court ruled there was contributory negligence. At the same time, it helps to find the common elements in the cases in which the court held there was no contributory negligence. With an extension of the routines used by Hunt and Hovland, one can think of a future computer program that looks at a large set of cases together with their legal conclusions to determine the common elements in the factual situations linking the facts to the conclusions.

C. *Resolving Rule Conflicts*

In heuristic programs it is common to find conflicts in rules and the decisions made from them. Heuristic programs must have some mechanism for handling conflicting decisions, because many if not all of the judgmental rules are neither logically complete nor internally consistent. Finding conflicts between principles underlying decisions is also an important aspect of legal reasoning. Unfortunately, little computer science research has been undertaken on this problem.

Currently most of the programs for which this is a problem, including Heuristic DENDRAL, avoid conflicts with a simple trick. They base their decisions on the first rule or "strongest" rule that applies to a given situation. Or, if the program can tolerate ambiguities, all the rules are applied, with a decision being made later about which consequences to accept. When Heuristic DENDRAL consults its rule table to formulate a plan, for instance, every relevant rule is applied. For this reason the program often formulates conflicting plans, all of which are explored.[26] One way the program resolves this ambiguity is to build models of each class and attempt to reduce the conflict later by testing predicted consequences of the specific instances of each plan. A second way it can resolve the conflict is to look at additional experimental data (if it is available) and apply additional rules to them.

A recent program written to allow the computer to learn to play draw

---

26. For example, the rules may indicate that both secondary amines and tertiary amines—two mutually exclusive chemical classes—should be generated.

poker uses the strategy of acting on only the first rule which is the strongest one that applies to a situation.[27] When this program is in its learning mode (when it is learning to play well), however, it has the capability of recognizing conflicts between its rules and reordering the rules or modifying them so that the first relevant rule in the table will be the correct one to apply. In learning, it builds a set of generalizations for good draw poker play from experience—either with or without advice from an expert (human or machine). When the program finds that the generalization it has just made conflicts with an existing rule, it resolves the conflict by either weakening or strengthening the rule. Exceptions to a rule can be noted, rules can be generalized by subsuming them under a broader rule, and overly general rules can be made more specific. For example, if the existing rule tells the program to bet high for three of a kind but its experience on the last hand has shown that three deuces were not worth a high bet, then it would amend its three-of-a-kind rule to exclude three deuces. Here there are straightforward methods for recognizing rule conflicts and eliminating them. Whether these methods apply in complex legal situations is an open question. An attempt to extend these methods into the legal problem-solving area might yield some interesting information concerning the ways in which lawyers and judges actually resolve rule conflicts.

D. *Finding and Using Analogies*

The fourth area where the legal reasoning program needs to build on computer research is in finding and using analogies. Here, too, the existing computer work has been in relatively simplified and formal domains. Analogical reasoning in law needs the richness of the English language, with great numbers of legal rules and factual situations cataloged in memory.

Computer work on finding and using analogies is rather sparse, in part because of the magnitude of the problems. One important program has been written to solve the common intelligence test problem of choosing a geometrical figure that exhibits the same relation to a given figure as two other given figures.[28] It finds the pair of figures, one given and one from a set of possible answers, that stand in the same relation to each other as a given pair of figures. The program has some built-in knowledge of the properties of geometrical figures and relations between figures. It searches for analogs among these concepts. This is a standard problem on IQ tests

---

27. D. Waterman, Generalization Learning Techniques for Automating the Learning of Heuristics (Stanford University Artificial Intelligence Memo AIM-102, July 1969) (available from the Clearinghouse for Federal Scientific and Technical Information, Springfield, Virginia).

28. Evans, *A Program for the Solution of Geometric-Analogy Intelligence Test Questions*, in SEMANTIC INFORMATION PROCESSING 271 (M. Minsky ed. 1968).

or entrance examinations, and was chosen partly for this reason, since it is so often used to discriminate intelligent problem-solvers from less intelligent ones.

Adding analogical reasoning to Heuristic DENDRAL in the future will be at least a two-step process. First, it should be able to use stored analogies, provided by a programmer, in order to solve new problems. After that, the program should be given the ability to find the analogies for itself.[29]

Discovering analogies in any field is a much harder problem than merely using them. Presumably the computer would be given a new problem and solutions to several old problems. It should be able to postulate some analogies and apply analogs of the known solutions to the new problem. As in the geometry program, the program would need some limited sets of concepts among which it could search for analogies.[30]

A legal reasoning program constrained to analogies along similar pre-specified dimensions would be excessively artificial. Many of the brilliant legal analogies between factual situations or principles are based on relations one probably would not have thought to put into a computer system. Yet, even a list of mundane relations, used in the same way as the geometric analogy program[31] uses geometric relations, could further the reasoning power of a program immensely.

VI. Conclusion

We have described a bit of what lawyers and computer scientists do, and speculated about a possible relationship between the two. Artificial intelligence may have something to contribute to law along this dimension. The only way of finding out whether the relationship will be fruitful is to explore it seriously.

It is premature to state categorically that computers will be used as aids in the process of legal reasoning, or even that they should be. It is hard even to imagine a consensus on the import of the research we propose. Certainly lawyers at one extreme will already have written off computers as a waste of time and money, while some at another extreme will be so convinced of the computer's potential as to feel threatened by its future encroach-

---

29. The analogies that it could use, and possibly find for itself, would be of the form "ethers behave like amines." If the program were able to carry on a dialog with the chemist about the aspects of amines and ethers for which the analogy holds true, it could build suitable ether rules. For instance, the rule table would give the program the four or five amine rules. It could postulate that exactly the same rules hold for ethers, and then modify them as a result of the dialog or further experience.

30. Knowing the higher order principle that analogies can be found between types of chemical atoms, for instance, the chemistry program could postulate that oxygen behaves like nitrogen. After checking this hypothesis among the known solutions it should be able to state firmly that ethers behave like amines.

31. See text accompanying note 28 *supra*.

ment upon their work. Between these extremes are numerous positions, some more plausible than others. Such a system could be developed only to die of neglect; it could survive only in the cloisters of academia; it could become an occasional tool of some small or large number of lawyers; it could, conceivably, become a major influence in the practice of law.

The research might lead to major changes in the role of law and lawyers in our society, to some changes, or to none. By removing the "dog work" that underlies quality legal service, the computer system might make good service available to a larger segment of society. Lawyers' conceptions of their role might change as they gain a wider understanding of the positions they advocate. Or the computer system might operate in such a narrow and restricted way that it would miss many possibilities that average lawyers now readily conceive. Lawyers might rely too heavily on a restricted, and thus somewhat incompetent, system with a resulting decline in the quality of legal services. The system might make it difficult to train lawyers for complex tasks requiring seasoned judgment developed through years of apprenticeship, because the apprentice's work had been replaced by machines. Or the system might be so widely available and provide such stable and usable information that ordinary people might become their own lawyers. Legal representation might no longer be needed in many disputes.

With courts and lawyers all using the same system, new stability in the law might be realized. This stability might help or it might hinder the overall development of the law and its ability to adjust conflict within society and to serve other ends. The adjustment of law to the computer might generate demands to rid law of its ambiguities and turn it into a logically consistent and complete system that would provide clear and unequivocal answers to all cases.

Control of the system and its programs might confer such significant power that battles over legislation would shift from the committees and floors of legislatures to the offices of programmers. The system might be so expensive that even the most affluent society could not afford it, or it might save so much legal labor that no society could afford to be without it. Because of its cost the system might only be available to the wealthy and would assist them in increasing their wealth. The success of the system might remove most of the needs for lawyers; its failure might attest to their value to society, or the research might show that even if the system were developed it would not help either the lawyers or their existing and potential clients.

It is possible to speculate, as many have, about an endless succession of utopias and horrors. The selection of any of these outcomes at this point can only be based on prejudice, hope, fear, and the like. Overcoming the

psychological, financial, and technological obstacles to this research is not a trivial task.

We have surveyed some computer programs that appear to attack non-legal problems in ways that lawyers attack their own. On this evidence, we have claimed that it would be worthwhile to explore the potential application of computers to legal reasoning. The programs developed in the course of research in artifical intelligence and cognitive simulation indicate a potential for computer performance of processes similar to those at times used by legal problem-solvers. These programs work with processes that form part of the lawyer's problem-solving apparatus, or at least, without benefit of more systematic study of this apparatus, so it seems. An attempt to extend or apply the programs to simulate the legal problem-solver should have at least these benefits: It should prompt more systematic study of legal problem-solving, and it should advance knowledge of the problem-solving capabilities of computers.

In short, research proceeding in computer science could enhance our understanding of the processes by which lawyers work and think. So far lawyers have not attempted to explore its relevance. They should.