Intelligent Computer Systems for Criminal Sentencing[1]

Uri J. Schild
Department of Mathematics and Computer Science
Center for Computers and Law
Bar-Ilan University
Ramat-Gan 52900, Israel
schild@bimacs.cs.biu.ac.il

Abstract: We discuss the construction of intelligent computer systems for sentencing decision support. We reject a rule-based approach, and argue that a case-based approach is acceptable. A prototype of a system of this kind has been constructed for actual testing in the Law Courts of Israel.

Keywords: Case-Based Reasoning, Legal Computer Systems, Sentencing

1. Introduction

1.1 The Problem

The problem described here deals with the challenge of creating a knowledge-based computer system that can assist judges in the process of passing sentence in criminal cases. The sentencing process calls for human discretion: Apart from mandatory sentences for specific offences, judges are usually faced with having to choose one out of many acceptable sentences. This process of selection and decision making is a most distinctive human task.

Most people - computer scientists, legal experts and laymen - do not aim for the development of computer systems that independently would carry out the tasks of a judge. They believe that as a matter of *principle* this activity should be an exclusively human one The goal should rather be to build a decision *support* system for sentencing. Such a system would supply a judge with relevant information, which it would obtain through an intelligent reasoning process using an expert knowledge-base. But the final decision would be made by the judge himself.

1.2 Practical Application

Our topic is not just a problem of interest for computer scientists working in the area of AI and Law, but can also have a major practical importance:

(1) It is well-known that the courts in many countries operate with a great backlog of cases. It is possible that the introduction of computer systems of the type described above may increase the efficiency of the human practitioners.

(2) It is universally assumed that a judge passing sentence in a given case should follow trends established in sentencing of similar cases. In practice, however, it appears that in many instances there is a great disparity in the decisions of sentencing judges - even in quite similar cases. It also happens that the very same judge will decide on vastly different sentences in similar cases occurring even over a relatively short time-span. There is therefore some dissatisfaction with the sentencing process as it takes place today. A computer system for sentencing support could possibly enable the judiciary to pass sentences of greater uniformity, without impairing their freedom and independence.

1.3 Acceptance by the Judiciary

We believe that the major principle of guidance in constructing a sentencing support system should be its ultimate acceptance and actual use by the judges. There would be no purpose in developing such a computer system, if it would be applied by only a very small number

[1] This work was partially supported by an Internal Research Grant from Bar-Ilan University's Research Authority

of judges, or not used at all.

In the following we shall assume that acceptance of a decision support system by the judiciary is the major criterion. Uniformity of the sentencing process will also be considered, but the question whether such computer systems would actually increase the efficiency of the courts will be ignored.

The outlay of this paper is as follows. In section two and three we shall consider work that relate to existing computer systems for sentencing support. In section four we shall describe the case-based approach we have adopted in our system, and section five will discuss and summarise the various approaches to sentencing advisory systems.

2. Previous Work: Non-Intelligent Systems

2.1 ASSYST

An extreme approach towards attaining uniformity in sentencing has been taken in the United States. The Sentencing Reform Act of 1984 is the statutory basis for the present federal sentencing guidelines. These guidelines are very specific, defining ranges of sentences covering felonies and some misdemeanors. If a sentence falls outside the prescribed range the offended party may appeal, solely on these grounds.

A computer system called ASSYST is based on these federal sentencing guidelines. It elicits from the user all the information required to make a decision, i.e. to determine a sentence according to the guidelines ([Simon89]). ASSYST is of no relevance to justice systems that do not subscribe to mandatory sentencing guidelines.

2.2 LIST

This system was developed at the University of British Columbia ([Hogarth88]). It takes a step in the direction of supplying a judge with relevant information, without actually prescribing a sentence. The database consists of sentencing decisions of the Provincial Court, Supreme Court and Court of Appeal of British Columbia. The system operates in the following way:

(1) The user selects the particular offence from a list of offences.

(2) He checks a small number of offender characteristics. For robbery these would be: age-range, use of weapon (yes/no) and past record of violence (yes/no).

(3) He may request a histogram providing sentence-ranges for all cases in the database which match the offender-characteristics of the case at hand.

(4) He may view all or some of the relevant decisions of the British Columbia Court of Appeal. These cases are retrieved according to the type of the sentence.

(5) The system permits the user to retrieve cases according to aggravating and mitigating factors relating to the offence, the record or the offender.

This approach has met with some criticism. The statistical knowledge embodied in LIST is based on a very small number of characteristics. This does not suffice to express the actual complexity of the sentencing process. To quote a paper dealing with computer aids for sentencing:

"Without being too imaginative, the following dimensions could be used to define 'criminal record': (a) the number of previous convictions, (b) the recency of the last conviction, (c) whether the past record includes violent offences, (d) the length of time since the offender first was convicted, (e) whether the present offence was more serious than the most recent offence he had been sentenced for, and (f) the nature or severity of the offender's most recent sentence. It does not take a mathematical wizard to realize that if there are even as few as three or four levels of each of these six variables, there are over 700 combinations of aspects of this one variable - criminal record." ([Doob87], p.61).

Imagine now that we want to take *all* possible variables and their levels into consideration. A judge would first have to determine the values of all those parameters in the present case. This calls for much discretionary effort, and may easily discourage the judge.

Furthermore, statistics would have to be compiled with respect to all those parameters. But the number of previous cases corresponding exactly in *all* parameters to the case at hand would be so small, that the statistical data would not be significant. Significant statistics could of course be compiled over a very long period. But this would be of little use: Sentencing data from, say, twenty years ago would probably not be relevant to the sentences being given today. Finally, detailed data relating to sentencing parameters is simply not available. We are not aware of any country where more than a few sentencing parameters are recorded and stored for future use.

The conclusion is, that the statistical data in LIST, while impressive, may simply be misleading in defining the starting point of the deliberations of a sentencing judge.

3. Previous Work: Rule-Based Systems

3.1 The System for Probation Officers

A rule-based system has been built in Israel for the use of probation officers in recommending sentences for young criminals ([Shapira90]). This system has been in existence and actual operation by the Youth Probation Services for several years.

The probation officer does not necessarily have to adopt the recommendation of the system, but he is required to give a detailed justification for any deviation from it. If the probation officer adopts the recommendation of the system, the judge does not have to follow this recommendation, though he usually does (in 82% of the cases: [Shapira90], p.161). The system has met with great success. It has been readily accepted by the probation officers, and is also said to save time and improve their efficiency.

Here we have an example of a knowledge-based system that actually assists in the sentencing process. However, it is important to consider the environment where this system operates. Probation officers are governmental employees who do not enjoy (and do not expect) the same kind of independence as judges. They must carry out orders and policies as formulated by their superiors with or without a computer system. If this policy happens to be expressed in rule-based form and included in a computer system, it makes their work easier, as experience indeed has shown. Furthermore, these probation officers are employed by the youth courts. Sentencing policy in those courts is much simpler than in courts for adults.

3.2 BAIL ADVISOR

Work is in progress in England on a rule-based system for bail-setting decisions ([Hasset93]). It appears that decisions on whether to release a suspect on bail lack uniformity. While one magistrate (judge) will deny bail to an arrested person, another judge will set some minimal amount of bail for a suspect detained under almost the same circumstances.

The English system under development uses the following approach. The magistrates are interviewed about the way they go about deciding about release on bail. This information is expressed in rule-form with the intention to build a rule-based expert system. When a magistrate has to make a bail-release decision he would query the system, which would supply him with a definite answer to the question whether to release or not to release the suspect. Obviously the magistrate would not be forced to follow the decision of the system, but hopefully this decision would be an important guide for him, when handing down his decision.

The applicability of this extremely interesting system is at present inconclusive ([Hasset94]). Also here it is important to consider the environment where the system is supposed to operate. Decisions on bail in the U.K. are made by the lowest level of judiciary, the magistrates. These are laymen of impeccable background, who have volunteered for this job. Thus, on one hand this group exhibit certain similarities with the probation officers, on the other hand the magistrates have many of the characteristics of professional judges.

3.3 Other Work

A small number of other projects have dealt with computer systems and sentencing, e.g. [DeMulder83], which uses a model-based approach. We shall not consider these any further, as the systems discussed above are representative for our purposes. A rule-based system has been developed in Tennessee for sentence calculation [Reynolds93]. It assumes that the judge has passed sentence, and computes the actual release dates for offenders. It is not of relevance to our problem.

[Berman89] surveys the problems related to computer systems for sentencing. The paper also discusses ways of actually implementing rule-based expert systems for sentencing, and raises the possibility of using MYCIN-like weights. MYCIN ([Davis77]) is one of the first medical expert systems developed. It associates a so-called certainty factor ([Johnson85]) with each rule and combines these factors according to the laws of fuzzy logic. [Berman89] proposes to associate 'danger factors' to rules in such a way that a "jail term may be indicated when the danger factor exceeds a particular value" (p.935).

4. The Case-Based Approach

4.1 Conceptual Retrieval

The term 'conceptual retrieval' has been used by several researchers, primarily by Hafner. It refers to a retrieval method which is based on the meaning and legal significance of the cases retrieved (as opposed to pure text retrieval). Given a conceptual retrieval system and a sentencing case-base, a judge about to determine a sentence could retrieve precisely those cases relevant to a stated concept.

According to [Hafner87] the knowledge base of a conceptual retrieval system consists of three parts:

(1) A domain knowledge model, which defines the concepts the system should understand and know about, both legal and common-sense knowledge. Also relations among these items of knowledge are included in the domain knowledge.

(2) Individual case descriptors, which are descriptions of each case in the case-database structured according to the concepts of the domain knowledge model.

(3) A hierarchical rule system (called the issue/case discrimination tree in [Hafner87]). This set of rules will enable the system to locate and retrieve the cases relevant to the problem at hand.

We have developed a computer system which uses conceptual retrieval in order to present relevant information from a sentencing case-base. The domain knowledge was elicited from an expert judge, the Vice-President of the Tel-Aviv District and Appeals Court, who has many years experience on the Bench, especially in criminal cases. After many sessions and iterations the parameters that judges take into account when passing sentence, were determined. These factors were also found appropriate by our academic associates, a professor of law doing research in the area of criminal law and a panel of criminologists.

The sentencing parameters were arranged as nodes in a set of discrimination trees. One tree deals with features relating to the offender himself. In that tree, e.g., 'not-main-offender' is a father-node of 'weak personality', which in its turn is the father node of 'easily-influenced by others'. Other trees deal with the victim, the crime itself, mitigating and aggravating circumstances. Other parameters relate to specific crimes.

It is obvious that the area of sentencing is associated with an enormous amount of both common-sense knowledge and domain knowledge. In fact, a complete model like the one proposed by [Hafner87] would be almost impossible to create, even if we limit ourselves to only a few specific paragraphs of the criminal code. On the other hand, a hierarchical structure is necessary for the system to carry out any kind of retrieval. We therefore decided to use the elicited domain knowledge *only*, without any additional common-sense knowledge. The nodes of the discrimination trees were taken as indices to cases in the case-base, i.e., these were the concepts a judge would be interested in.

We have limited ourselves to two serious crimes:

Robbery and Rape. As indexing of cases is a technically big problem for large case-bases, we decided to establish a case-base spanning only the last five years. A larger time-span would raise problem of trends and changing attitudes in sentencing, which at this stage we do not intend to deal with. In Israel robbery and rape cases are heard in the District Courts, and may be appealed to the Supreme Court. For reasons to become apparent in the next section, we selected only Supreme Court cases, and only those where the appeal relates to the actual sentence. The prototype case-base covering the last five years contains less than a hundred cases, and manual indexing was thus possible. It was carried out by a qualified criminologist and a graduate computer science student, who had also been involved in the knowledge elicitation process.

The system assists the user to walk through the discrimination trees and check the nodes that are relevant to his case. The system then retrieves those cases from the case-base, which are indexed by the chosen nodes. Furthermore, retrieval is also carried out for nearest neighbours of the chosen nodes. Thus, if the user has checked the node 'not main offender' in the offender-tree, the system may also retrieve cases indexed by a sister-node, e.g., 'assisting after execution of crime'.

Obviously cases retrieved by neighbouring nodes have less bearing than cases which exactly fit the checked parameters. The system uses a similarity metric to order the retrieved cases before presenting them to the user. For each case the system also presents a list of the relevant parameters. The user can choose three levels of output: (1) a formulation of the ratio of the case, (2) the parts of the law-report relevant to the sentencing, (3) the entire law-report.

The expert knowledge of the system is contained in its discrimination trees, in its capability of retrieving not only directly relevant cases, but also cases indexed by neighbouring nodes, which may have some bearing upon the new case. The system was presented to several judges, who found it of far greater use than the classical text retrieval systems, but of limited application because of the small case-base.

The system could have been further developed in several directions. Using Hafner's ideas much more intelligence could be added, and the case-base could be significantly enlarged. We did not proceed with any further development of this system, as we decided to use it not as a stand-alone system, but as a complement to the system described in the next section.

4.2 Case-Based Reasoning

4.2.1 Case-Based Explanation

Case-Based Reasoning (CBR) is a problem solving approach by humans and computers where: "new problems are approached by remembering old similar ones and moving forward from there"([Kolodner93], p.xiv). Given a new problem, a CBR program retrieves relevant cases from a case-base (using appropriately defined indices), chooses the most similar case and adapts its solution to the new problem. Conceptual retrieval, as we considered it in the previous section, is the first step of CBR.

The use of previous cases is a central aspect of legal reasoning in general, and in the area of sentencing in particular. When a judge has to pass sentence in a new case, he often considers old and similar cases and may adapt an old sentence to fit the new case. These old cases are not only the (perhaps binding) precedents he must consider, but primarily cases from his own experience, or cases his colleagues have told him about. A judge we interviewed told us how he actually keeps a card-index of his old cases and their sentences, which he flips through when he has to pass sentence in a new case.

Several researchers have previously applied methods of CBR to the legal domain, but not to the area of sentencing (see, e.g., [Rissland87], [Ashley90], [Branting91]), [Skalak92]. [Ashley92] is a most comprehensive overview of the use of CBR in Law.

An early CBR program was JUDGE ([Bain86]). It used the sentencing domain to test cognitive theories of reminding and problem-solving. It was not intended as a

program of practical use by judges. Our experience from interviewing judges closely correspond to the cognitive observations in Bain's work (see [Riesbeck89]).

The question arises, which cases a judge is reminded of, when determining the sentence of a new case. A case usually has one or more special and outstanding features. In the CBR community such a feature is called an 'anomaly' of the case, though its appearance may be quite ordinary. When encountering such an anomaly in a new case, a judge may be reminded about an old case with the same anomaly. Thus the anomalies could form the indices for a sentencing case-base.

In order to discover what the anomalies could be, we interviewed judges from the Tel-Aviv District and Appeals Court. We did *not* include the judge who had previously been involved in knowledge elicitation. The interview consisted in asking the judges to tell us about old cases of theirs, in the two areas of the criminal law we are concentrating on, Robbery and Rape. It then appeared that when a judge told us about a case, he would invariably come up with a *heading* for this case. He would supply that heading unasked, before or after telling about the case and sometimes in the middle. This heading turned out to be the most prominent anomaly of the case.

We shall give a concrete example of this. Consider the following story told by a judge:

A young man had one night been out driving with a friend of his. The friend had then suggested they should rob a gas station. The man really had no intention of getting involved, but his friend eventually made him come along. They were later apprehended and found guilty. The man had unfolded the entire story in court, making a rather honest impression.

The judge explained his way of determining the sentence in this case, and ended by remarking: "That's what happens when you cannot say no".

It became apparent that the anomalies supplied by the judges were closely corresponding and sometimes even identical to the sentencing parameters (i.e. discrimination tree-nodes) described in the previous section, thus confirming those structures. The judge's remark in the above story we constructed to mean, that 'easily influenced by others' (a node in the offender's discrimination tree from the previous section) would be an index to the case.

Having determined the indices, our idea was to consider the sentencing process as case-based explanation ([Schank94]). We shall call the sentence of a case the 'explanation' of the case. Determining the sentence of a new case means finding a similar old case and using its 'explanation' (adapting its sentence) to explain the new case.

It often happens that a judge will decide upon a sentence, and only afterwards attempt to justify it - to himself or to others. One judge expressed this phenomenon by saying that he has a 'gut-feeling' of what the sentence should be. Obviously this justification is also an 'explanation' in the ordinary sense of this word.

4.2.2 Knowledge Representation and Case-Base

Knowledge about explanation of simple cases may be represented by an Explanation Pattern (XP) ([Schank94]). However, legal cases are complex, and deciding a sentence necessitates the weighing of several factors. We have therefore created a more comprehensive structure, a Multiple Explanation Pattern (MXP), which is defined as a *collection of viewpoints* relating to the sentence in the same criminal case. Each such viewpoint relates to a fact that contributes to (increases or decreases) the sentence.

Each viewpoint is represented by an XP, and carries a weight relative to the other viewpoints. It is crucial to understand that an XP as such has no independent existence. Only the totality of the XPs forming the MXP describes the particular case and its sentence. We shall present *part* of the MXP for the robbery case briefly described above.

basic facts:

accused according to paragraph 402
found guilty
baseline sentence: 4 years
maximal sentence: 20 years
actual sentence: 1 year
appealed: no

XP-1:

facts: first offence
beliefs: not dangerous to public
purpose: retribution, rehabilitation
plan: strong mitigation
action: reduce baseline sentence
weight: above medium

XP-2:
facts: confessed
beliefs: seems trustworthy
purpose: retribution, rehabilitation
plan: weak mitigation
action: reduce baseline sentence
weight: light

XP-3:

facts: easily influenced by others
beliefs: not dangerous to public
purpose: retribution, rehabilitation
plan: extreme mitigation
action: reduce baseline sentence
weight: heavy

Each XP has the following slots:

1) Facts: This slot contains an index, which is a *leaf* in the index-hierarchy.
2) Belief: The values of this slot are (intermediate-level) indices in the hierarchy.
3) Purpose: This slot contains one or more of the four sentencing approaches.
4) Plan: Depends on the value of the purpose slot, e.g., if rehabilitation is the sentencing purpose, then the plan-slot could indicate some degree of mitigation.
5) Action: The value here indicates the action to be taken with respect to a baseline sentence.
6) Weight: Each XP in an MXP has a fuzzy-set value in this slot, indicating the importance of the particular XP in relation to the other XPs in the given MXP.

All slots and basic facts of the MXP are filled out for old cases under the guidance of the sentencing judge. The sentence measured out in a case reflects the combinations of XPs in the MXP, though no numerical formula for this combination is possible. We are at present experimenting with different combinations of the fuzzy-set values, which must be acceptable to the judges.

The basic facts of a retrieved old MXP contains information whether the old case was appealed. The user may retrieve relevant precedents from the Supreme Court by applying the conceptual retrieval component described in the previous section. This is enabled by simply clicking on any of the facts-slot of the XPs forming the MXP.

Given a new case a MXP-skeleton must first be created. It is then natural to retrieve an old case with a similar MXP, adapt the old MXP to the new case, and correspondingly change the old sentence to become a suggested new sentence. Normally, the system will retrieve more than one old case, and therefore suggest several sentences. However, if the case-base is uniform, the suggested sentences will be within a small range.

Several judges have emphasized, that the chosen sentencing approach (rehabilitation, retribution, prevention or deterrence) is a key parameter. The user is therefore requested to state which sentencing approach he is considering, and the system will execute the retrieval and adaptation only of MXPs with the same sentencing approach found in its major XP. As the system is interactive, a judge may experiment with several sentencing approaches, view the suggested sentences of each approach and only then make up his mind and come to a decision.

One of the basic facts is labeled as 'baseline sentence'. This is the baseline chosen by the particular sentencing judge. Different judges will often chose different baselines, and the program must of course take this into

account, and make an appropriate adjustment.

4.2.3 Retrieval, Adaptation and Application

It is not our intention to give detailed descriptions and algorithms for the retrieval of old MXPs, choice of the 'best' one and its adaptation to fit the MXP of a new case, including a proposal of a sentence or a sentencing-range for the new case. We are arguing in this paper for the case-based paradigm to sentencing, but not necessarily for the particular approach we have chosen.

Some problems have come up for which our solution is not necessarily the correct one. Only the actual experience by judges with the system will settle those problems. We shall here give two examples of problems, that have no clear solution:

(1) It may happen that the best retrieved case has a feature (i.e., an XP), which does not appear in the new case. Conversely, the new case may have a feature, that does not appear in the best retrieved case. In the first instance we delete the XP and renormalize the fuzzy values. In the second instance we have collected a set of 'generalized' XPs for use in such situations.
(2) The use of fuzzy-set values (very heavy, heavy, etc.) is just a way of hiding explicit numerical computations. Such computations must of course be carried out, but there is more than one way of doing this.

5. Discussion

Several approaches to sentencing advisory systems have been considered here. We shall now summarise our findings.

(1) Statistical Information Systems.

We have seen that such systems are based on an insufficient number of parameters. They may actually be misleading, as users who are not expert statisticians or computer scientists, often trust facts and figures supplied by a computer, and accept them as significant.

(2) A Sentencing Guidelines Program.

This approach has met with vehement opposition from the judiciary in the USA. The opposition is of course to the principle and not to its computer implementation. The large measure of cooperation we have enjoyed from the judiciary is to some extent in order to improve any acceptable alternative to sentencing guidelines.

(3) Model-Based Systems.

Research in model-based expert systems has been promising in many areas. However, our judiciary has no confidence in criminological sentencing models (see [Love89]), and totally reject this approach.

(4) Rule-Based Advisory Systems.

The practical usability of such computer systems in various domains has been under discussion for over twenty years, with opinions ranging from total rejection to total acceptance. We have surveyed two such systems above. The first of these has had a proven success, the second is still under development.

Both these two systems deal with decision problems similar to criminal sentencing. However, the persons making those decisions are not professional judges. In the Israeli system they are social workers (youth probation officers), and must necessarily follow the regulations of their service. In the British case the lowest level of judiciary is made up of non-professional judges. They do not have to follow any strict regulations imposed from above, and thus have the independence of judges. It is possible that they will agree to work with a rule-based computer system.

This, however, is not the case with professional judges. Our impression from speaking with judges is one of total rejection of the idea of using fixed rules for sentencing. The rules of a sentencing system would reflect the private view and opinion solely of the judge formulating the rules. These opinions may not necessarily be held by other judges. A panel of judges may not reach an

agreement about sentencing rules, and even if they did, no judge would be obliged to follow those rules, or even to consider them while passing sentence.

We may add some further reasons why the rule-based approach is not appropriate for sentencing advisory systems. These reasons relate to rule-based expert systems in general, and have been observed many times in the past ([Wellbank83]).

i. Experts are often inconsistent in the sense that they do not practice what they say they do. In other words, even if they are willing to formulate rules, they do not always follow those rules themselves.

ii. Rules extracted from a panel of experts are often conflicting. Even a single expert often contradicts himself. In the case of judges these inconsistencies may reflect the actual inconsistent sentencing.

iii. A prominent feature of rule-based systems is their capability of explaining the results they supply when queried. However, the systems do not enable differentiation between rules of a technical nature and rules with conceptual expert knowledge, so the explanation of results becomes just an explanation of the formal deductions.

iv. Extracted rules have semantic vagueness. It is not clear how they should be formalized in a computer program. Whatever formalization is decided upon, implies that discretion is ignored.

(5) Conceptual Retrieval Systems

Our experience in developing such a system, and its acceptance by the judges has shown us the advantages of this approach. The main problem is the indexing of the case-base, which could be very large. Much research is being done at present on automatic indexing using methods of Natural Language Processing. Even if this research bears fruit, conceptual retrieval will be only a first step, as it stops before the actual reasoning.

(6) Case-Based Advisory Systems

Cases represent an experienced situation. When a similar situation arises, those decisions and the knowledge that went into making them provide a starting point for solving the problem it poses. In other words, using the CBR approach it will be possible for a judge to determine a sentence based on general standards but also to consider the individual circumstances of the offender and the case at hand.

Judges are accustomed to work with cases, to apply them and to distinguish them. As we have described above, a case-based advisory system presents the judges with real cases and sentences, not made-up rules. If judges are at all ready to use a computer for sentencing support, cases is the natural media for conveying information.

Cases in a case-base may appear to be conflicting, just as rules may conflict. There is however a major difference between these two kinds of inconsistencies. Cases in the case-base reflect *real legal situations*, and legal experts are accustomed to resolve conflicts in case-law. They also deal with inconsistencies in statutory law, but the sentencing rules in a rule-based expert system are not the law, they have no legal standing. They are superficial creations, that carry no more weight than the individual judge decides to assign them (which may be none at all). If such rules were made binding by legislation, they would actually be equivalent to the sentencing guidelines mentioned in section 2.1, and the judiciary wants to avoid that at all costs.

One problem must be considered: Which cases should be included in a case-base for sentencing. We plan to deal with this question in two steps. A judge may first be willing to include only his own previous sentencing decisions in what will then become his own private case-base. This would supply him with a computerized form of the card-index we mentioned above. It would not contribute to the aim of attaining general uniformity in sentencing, but at least the judge's own sentences may become uniform. The second step would be to select (and maintain!) a case-base by a public panel, perhaps consisting of judges, respected lawyers and academics. Our present aim is simply to present our prototype CBR system to judges, receive some feedback and improve the system.

6. Conclusion

We have presented some approaches to advisory computer systems for criminal sentencing. It is our opinion that among those approaches the case-based one is the only feasible one. This takes into account the objective problems of other kinds of systems, and the opposition from the judiciary itself to these other kinds of systems. The introduction of our present prototype on a limited trial basis in the Israeli courts will test this hypothesis.

7. Acknowledgements

The conceptual retrieval program was written by Gil Erlich and the CBR program is by Yaakov Kerner.

8. References

[Ashley90] Ashley K.D. Modeling Legal Argument MIT Press, Cambridge, 1990

[Ashley92] Ashley K.D. Case-Based Reasoning and its Implications for Legal Expert Systems Artificial Intelligence and Law, 1, 1992, p.113-208

[Bain89] Bain W.M. Case-Based Reasoning: A Computer Model of Subjective Assessment Ph.D. Thesis, Yale University, 1986

[Berman89] Berman D.H., Hafner C.D. The Potential of Artificial Intelligence to Help Solve the Crisis in Our Legal System. Communications of the ACM, 32, 1989, p.928-938

[Branting91] Branting L.K. Reasoning with Portions of Precedents. 3rd Int. Conference on AI and Law, Oxford, 1991, p.145-154

[Davis77] Davis R., Buchanan B., Shortliffe E. Production Rules as a Representation for a Knowledge-Based Consultation Program. Artificial Intelligence, 8, 1977

[DeMulder83] DeMulder R.V., Gubby H.M. Legal Decision Making by Computer: An Experiment with Sentencing. Computer / Law Journal, IV, 1983, p.243-303

[Doob87] Doob A.N., Park N.W. Computerized Sentencing Information for Judges Criminal Law Quarterly, 30, 1987-88

[Hafner87] Hafner C.D. Conceptual Organization of Case Law Knowledge Bases. First Int. Conference on AI and Law, Boston, 1987, p.35-42

[Hasset93] Hasset P. Can Expert System Technology Contribute to Improved Bail Conditions Int. J. of Law and Information Technology, vol.1, p.144-188

[Hasset94] Hasset P. Private Communication.

[Hogarth88] Hogarth J. Computer and the Law: Sentencing Database System, User's Guide

[Johnson85] Johnson L., Keravnou E.T. Expert Systems Technology. Abacus Press, 1985

[Kolodner93] Kolodner J. Case-Based Reasoning Morgan Kaufman Publishers, San Mateo, California, 1993

[Love89] Lovegrove A. Judicial Decision Making, Sentencing Policy and Numerical Guidance Springer Verlag, Berlin, 1989

[Reynolds93] Reynolds D., Beck T, Tennessee Offender Management Information System AI Magazine, vol. 14, no. 3, Fall 1993, p.61-68

[Riesbeck89] Riesbeck C.K., Schank R.C. Inside Case-Based Reasoning Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1989

[Rissland87] Rissland E.L., Ashley K.D. A Case-Based System for Trade Secrets Law 1st Int. Conference on AI and Law, Boston, 1987, p.60-66

[Schank94] Schank R.C., Kass A., Riesbeck C.K. (editors) Inside Case-Based Explanation Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1994

[Simon89] Simon E., Gaes G. ASSYST - Computer Support for Guideline Sentencing. 2nd Int. Conference on AI and Law, Vancouver, 1989, p.195-200

[Shapira90] Shapira M. Computerized Decision Technology in Social Service Int. J. of Sociology and Social Policy, vol.10, 1990, p.138-164

[Skalak92] Skalak D.B., Rissland E.L. Arguments and Cases Artificial Intelligence and Law, vol.1, p.3-44

[Wellbank83] Wellbank M. A Review of Knowledge Acquisition Techniques for Expert Systems. British Telecom, Ipswitch, 1983