Computer Methods and Programs in Biomedicine, 30 (1989) 185–198
Elsevier

COMMET 01033

Case-based tutoring from a medical knowledge base

Homer L. Chin [1] and Gregory F. Cooper [2]

[1] *The Permanente Medical Group, Inc., San Jose, CA 95119-1197, U.S.A., and* [2] *Medical Computer Science Group, Stanford University Medical Center, Stanford, CA 94305-5479, U.S.A.*

The past decade has seen the emergence of programs that make use of large knowledge bases to assist physicians in diagnosis within the general field of internal medicine. One such program, Internist-I, contains knowledge about over 600 diseases, covering a significant proportion of internal medicine. This paper describes the process of converting a subset of this knowledge base — in the area of cardiovascular diseases — into a probabilistic format, and the use of this resulting knowledge base to teach medical diagnostic knowledge. The system (called KBSimulator — for Knowledge-Based patient Simulator) generates simulated patient cases and uses these cases as a focal point from which to teach medical knowledge. This project demonstrates the feasibility of building an intelligent, flexible instructional system that uses a knowledge base constructed primarily for medical diagnosis.

Intelligent computer-aided instruction; Medical computer-aided instruction; Medical knowledge base; Patient simulation

1. Introduction

One of the difficulties students have in learning medical diagnosis is adapting what they have learned in the classroom to the diagnosis of patients. The proliferation of factual knowledge within each medical specialty has led to increased rote learning and to a lack of experience with integrating such knowledge for diagnosis and problem solving. A project panel on the general education of the physician reiterated what many others have said before: medical schools should reduce their dependence on lectures as the principal method of teaching, and should increase activities that provide students with more opportunities for independent learning and problem solving [29]. One way to increase opportunities for problem solving is to use the computer to provide the student with simulated patients to diagnose and manage [15]. A major stumbling block to the further development of this approach to medical education, however, has been the costly and time-consuming process of creating didactically useful patient cases for computer-based education [21]. One way to overcome the high cost of creating patient cases for computer-based teaching is to make use of large existing medical knowledge bases, such as Internist-I/QMR [28], HELP [33], RECONSIDER [4], and DXplain [2],to generate didactically useful teaching cases automatically. The knowledge base can also be used to tutor the student flexibly, according to the student's specific needs. In addition, the knowledge base can be used to model the student's knowledge, so that specific tutorial interaction can be tailored to address deficiencies in the student's own fund of knowledge.

Correspondence: Homer L. Chin, M.D., The Permanente Medical Group, Inc., 260 International Circle, San Jose, CA 95119-1197, U.S.A.

2. Background

2.1. Intelligent tutoring

The primary feature of an intelligent tutoring system is that the system itself has *knowledge* about the subject matter that it is tutoring: the system is able to solve the same problems that it is presenting to the student. By making use of this knowledge base, the system is able not only to tutor the student about the specific solution to the problem, but also to use this knowledge to diagnose the student's errors and to tailor teaching strategies and material to the student's specific deficiencies. Because of its knowledge about the general subject matter, the system may also be capable of engaging in a meaningful *dialogue* with the student in areas that may not be directly related to the specific problem at hand [5]. In general, intelligent tutoring systems must have a knowledge base from which they generate or solve problems, problem-solving expertise, diagnostic or student-modelling capabilities, and some explanation ability [35].

Considerable research in the use of knowledge bases for intelligent tutoring has been done in the context of the GUIDON/NEOMYCIN projects [10]. GUIDON [8,9] is an intelligent computer-aided instructional (ICAI) system that makes use of the MYCIN [34] rule base to teach students the rules that MYCIN used to diagnose and recommend therapy for cases of bacteremia and meningitis. By interfacing a separate tutoring knowledge base that contained rules about tutoring strategy, GUIDON was able to tutor the student about the rules, or *heuristics*, contained in the original MYCIN knowledge base. MYCIN and many other rule-based expert systems, however, use an implicit nonpsychological strategy for diagnosis; these strategies, although sufficient for diagnosis, often are of limited value for teaching. MYCIN contains neither knowledge about the structure and strategy of medical diagnosis, nor support knowledge (the underlying justification for a rule) [10]. Subsequent work by Clancey has focused on augmenting and restructuring the knowledge base to incorporate rules for diagnostic strategy and increase its usefulness for teaching.

In this project, we make use of a probabilistic knowledge base to teach *associational* rather than *heuristic* knowledge. Since the knowledge that we are teaching is that of associating manifestations with diseases, rather than on the larger task of 'how to do diagnosis', we have bypassed some of the problems being addressed in the GUIDON/NEOMYCIN projects by focusing on a smaller teaching goal. We are not teaching the student how to diagnose, but rather, we are teaching the association between findings and diseases — that is, the differential diagnosis of findings. We wish to teach the student what are the possible diseases given a manifestation or set of manifestations, and what is their probability of occurrence.

2.2. Case-based instruction

Although GUIDON used the MYCIN knowledge base from which to tutor it did not generate patient cases from this knowledge base; rather, it used cases selected from a prespecified library of patient data. In contrast to this, the KBSimulator system described in this paper makes use of patients *generated de novo* from the knowledge base.

GUIDON supplements the MYCIN rule base with a separate rule base containing approximately 200 tutorial rules that guide the system in a tutorial dialogue with the student. In addition, the student is able to query the MYCIN knowledge base about how and why certain conclusions were reached. In a somewhat analogous manner, KBSimulator presents a patient case for the student to diagnose, and allows the student to explore the knowledge base on her own initiative to answer questions that she might have while diagnosing a particular patient case.

2.3. Patient-case generation from a knowledge base

De Dombal used his abdominal pain diagnostic system to simulate patients for teaching purposes [13]. Thirty-five findings were recorded from a series of 600 patients who were suffering from one of six different abdominal diseases. From these data, a table was produced linking the frequency of the occurrence of a particular finding to each disease. Using this table and a random-number generator, the system generated simulated patients who had one of the six encoded diseases, according to the frequency of occurrence of each finding in a patient with that disease. Although there is no published discussion of why this work was not pursued further, our research suggests that the approach used by de Dombal may lead to significant problems because the knowledge base does not encode relationships among the findings within a disease process (i.e., it assumes conditional independence) [6]. Because the presence of one finding may make the presence of another finding highly unlikely or impossible (e.g., for the manifestations MALE and PREGNANT, the presence of one excludes the possibility of the other), the lack of representation of this dependency can lead to the generation of patient cases with unlikely combinations of findings (such as a patient who is both MALE and PREGNANT).

A similar approach to de Dombal's was used, adapting the Internist-I knowledge base for the purpose of patient-generation, in the TEST program [31]. However, similar problems arising from inadequately encoded dependencies among patient findings led to the generation of patients with inconsistent findings. A project to augment the Internist-I knowledge base with causal knowledge and to use this augmented knowledge base to create simulated patient cases was explored in the CPCS project [27,31]. Although this augmented knowledge base was capable of generating clinically consistent patient simulations, it was estimated that it would take an order of magnitude greater effort to convert the existing Internist-I/QMR knowledge base into this augmented knowledge base than has already been expended in the 25 person-years of the development of the existing Internist-I/QMR knowledge base.

3. Overview of the KBSimulator system

Fig. 1 shows a diagram of the KBSimulator system. We converted a subset of the Internist-I knowledge base, in the domain of cardiovascular diseases, into a probabilistic format, which we used as the knowledge base. This subset contains knowledge about 31 cardiovascular diseases and their relationship to over 350 manifestations.

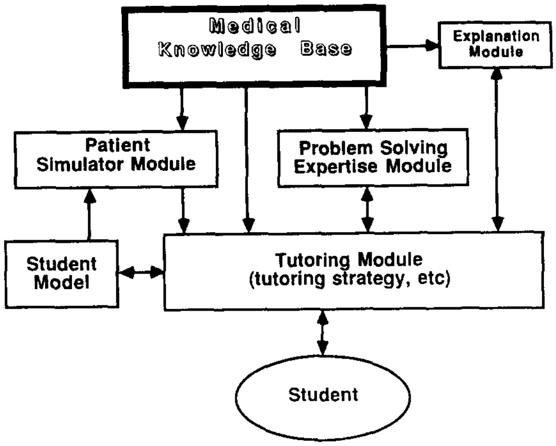

Fig. 1. The KBSimulator system, illustrating the various components of the system.

As shown in Fig. 1, the *medical knowledge base* serves several purposes: it is used to generate patient cases, it can be queried for information by the student, it is used to 'solve' the patient case that is being presented to the student, and it is used as a framework for rudimentary explanation.

The system generates simulated patient cases directly from the knowledge base and places these patient cases into a 'lesson'. A tutoring subprogram presents this patient case to the student, one manifestation at a time. As manifestations are presented to the student, the system applies its own expertise, using Bayes' rule, to calculate the probability of each possible disease given the manifestations presented up to that point. When the probability of some disease reaches a predefined threshold, the presentation of manifestations is halted, as the student has presumably been given sufficient information to diagnose the case. At this point, the student engages in a mixed-initiative interaction with the computer system, and is given the opportunity to make the diagnosis, to obtain more information about the patient, to examine the knowledge base for the differential diagnosis of any of the manifestations in this patient, or to ask the computer to explain which manifestations are the key ones in making the diagnosis. As the student interacts with the program, the computer keeps track of her performance. This rudimentary *student model* is used to direct the generation of subsequent patient cases.

4. The KBSimulator knowledge base

We decided to create a program that initially teaches students about cardiovascular diseases, both because the pathophysiology of cardiovascular diseases is reasonably well understood, and because it is an important general area in medicine and thus could form the basis for a system of benefit to all medical students, no matter what their specific interests.

We used 31 of the 47 cardiovascular diseases known to the Internist-I system, eliminating those that we thought were relatively clinically unimportant.

To build a knowledge base that had manifestations that would be optimally useful for teaching, we eliminated manifestations that either occurred too infrequently or were too nonspecific to be of much diagnostic use. In this way, nonspecific manifestations or relatively rare manifestations of limited specificity, such as AFFECT APPREHENSIVE and HICCUPS, were eliminated from the knowledge base.

Pruning the knowledge base resulted in a decrease in the number of manifestations from 600 to about 450. The resulting list of manifestations and diseases was examined by our collaborating cardiologist, and revealed many more manifestations that he deemed relatively unimportant for teaching purposes. In addition, our expert replaced several manifestation and disease terms with terms in more current usage; for example the term PYROGENIC SHOCK was replaced with SEPTIC SHOCK. The additional elimination of manifestations brought the total number of manifestations used in our knowledge base down to about 350. The final knowledge base used by KBSimulator therefore contained knowledge about 31 cardiovascular diseases and their relation to 350 manifestations.

4.1. Probabilistic conversion from Internist-I

4.1.1. Probability and Bayes' rule

The earliest knowledge bases that were successfully applied to medical diagnosis were programs that used Bayes' rule to calculate the probability of a disease given a constellation of patient findings. By making the assumptions that the diseases under consideration are mutually exclusive and exhaustive, and of the conditional independence of manifestations given a disease, it is possible to apply Bayes' rule over a set of manifestations, thereby calculating the probability of a particular disease given a combination of findings.

Examples of diagnostic systems that use Bayes' rule in this manner are Warner's program for the diagnosis of congenital heart disease [36] and de Dombal's program for the diagnosis of abdominal pain [14]. Although the assumption of mutually exclusive and exhaustive diseases and the assumption of conditional independence of manifestations given a disease are rarely satisfied completely in real life, the performance of some of these systems within limited domains has been impressive [1].

4.1.2. Ad hoc reasoning systems

Several ad hoc knowledge-representation and diagnostic-reasoning schemes have also been developed for computer-assisted medical diagnosis. The Internist-I system in particular uses subjective weights, called frequencies and evoking strengths, to link diseases to findings, and uses disease-independent/finding-specific imports that represent the degree to which one is compelled to explain the presence of a given finding in a patient [26]. The scoring scheme used is also limited to the Internist-I system. For example, the statement 'the score for diagnosis X is 130' is specifically based on the numerical weights in the Internist-I system and is not readily translated to other uncertainty-representation methods.

AI researchers have often used ad hoc methods for reasoning under uncertainty, citing limitations of probability theory as their justification for doing so. Further research, however, has begun to elucidate the correspondence between some of these ad hoc methods and probability theory [18,19,22]. Some of the advantages of using probability theory over an ad hoc method are that assumptions made in probabilistic reasoning systems are explicit, probability theory is widely understood, knowledge bases that make use of this theory could be adapted by different researchers to their own purposes without the need for extensive interpretation and validation, and the use of probability theory makes it possible to incorporate statistical data into the knowledge base and to validate the knowledge base with statistical data. For these reasons, we decided to convert our subset of the Internist-I knowledge base into a probabilistic form to take advantage of the properties of such a knowledge-representation scheme.

4.1.3. Mapping frequency numbers into probabilities
Earlier work done on the Internist-I knowledge base explored the correspondence between the Internist-I concepts of evoking strength and the Bayesian notions of predictive value and belief updates [18]. Studies have shown, however, that physicians are more reliable in giving the probability that a patient with a particular disease will exhibit a particular symptom, than they are in giving the probability that a patient exhibiting a particular symptom will have a particular disease [24]. For example, physicians can more accurately estimate the probability that a patient with a pulmonary embolus will be short of breath than they can estimate the probability that a patient who is short of breath has a pulmonary embolus. For this reason, we decided to focus on the use of the Internist-I frequency numbers (which would be more accurate than the evoking strengths) to accomplish our knowledge-base conversion.

Since the Internist-I frequency numbers are a measure of sensitivities, that is, $P(Mx|Dx)$ [26], we converted these numbers directly to their corresponding probabilities. This was done by obtaining probability conversion tables from R. Miller, one of the primary developers of the Internist-I system. On two separate occasions, estimates of the general correspondence between $P(Mx|Dx)$ and the Internist-I frequency measures were obtained. On the first occasion a point probability was obtained, and on the second, a probability interval was obtained. These correspondences are shown in Table 1. The elicited point probabilities showed a close correspondence to the midpoint of the probability intervals, and we decided to use the point probabilities for our knowledge-base conversion.

TABLE 1

Probability estimates of Internist-I frequency numbers

Frequency	Probability interval (%)	Point probability (%)
1	0– 5	2.5
2	5– 36	20
3	37– 64	50
4	65– 95	80
5	95–100	98.5

4.1.4. A priori probabilities
We obtained a priori probabilities for each of the 31 diagnoses by mapping them into their ICD-9 equivalents and obtaining the number of admissions to U.S. hospitals for those codes for the year 1983, the last year for which we were able easily to obtain such statistics [23]. In those cases for which there was no equivalent ICD-9 code corresponding to an Internist-I diagnosis, we used an estimate of the number of admissions with that diagnosis in one year. By dividing the number of admissions for a particular diagnosis by the number of total admissions for the 31 diagnoses, we obtained the a priori probability for that diagnosis (i.e, the a priori probability of *that* diagnosis given that the patient has one of the 31 diagnoses). In doing so, we make the 'completeness' and 'mutual exclusivity' assumptions; that is, we assume that the patient has one, and exactly one, of the 31 diseases.

4.1.5. Baseline manifestation occurrence and import
Internist-I contains frequency and evoking strength numbers linking those diseases and manifestations that are positively associated with one another, but it does not record a number for those diseases and manifestations for which there is no 'greater than baseline' association. For example the manifestation PAIN RIGHT BIG TOE is not associated with the disease ATRIAL SEPTAL DEFECT, and therefore the Internist-I disease profile for the disease ATRIAL SEPTAL DEFECT would not contain the manifestation PAIN RIGHT BIG TOE, and there would not be an explicit frequency number linking the two. However, the probability of finding the manifestation PAIN RIGHT BIG TOE

TABLE 2

Missing sensitivity numbers after mapping frequency into corresponding sensitivities

Prior probability		Mx1	Mx2	Mx3	Mx4	Mx5	...	Mx350
0.01	**Dx1**	x	?	?	x	?	...	?
0.20	**Dx2**	?	x	x	?	?	...	x
0.03	**Dx3**	?	?	?	x	?	...	?
0.18	**Dx4**	x	?	?	?	x	...	?
.	.	.	.	.	.	.	...	.
.	.	.	.	.	.	.	...	.
.	.	.	.	.	.	.	...	.
0.12	**Dx31**	?	x	?	?	?	...	x

in a patient with ATRIAL SEPTAL DEFECT is not zero (although it is quite small). In other words, for a large majority of manifestation-disease pairs, there is no explicit encoding of frequency, as there is no 'greater than baseline' association between that manifestation and that disease.

After mapping frequency into corresponding sensitivities and obtaining prior probabilities as described in the previous sections, we obtain the sparse table depicted in Table 2, where the x's signify frequencies that are encoded in the knowledge base, and ?'s signify those manifestation-disease pairs for which there is no explicit number encoded, and for which we must assume some 'baseline occurrence' of that manifestation. As these baseline occurrences are presumably disease independent, only one probability estimate is needed for each manifestation. Thus, for our KBSimulator cardiovascular disease knowledge base, only 350 such numbers would be needed: one estimate of 'baseline occurrence' for each manifestation.

Internist-I assigns to each manifestation a disease-independent *import number*, which ranges from 1 to 5, defined as the 'global importance of the manifestation' — that is, the extent to which one is compelled to explain the manifestation's presence in any patient [26].

The exact probabilistic definition of *import* is a matter of continuing research. However, there seems to be a strong correspondence between import and the baseline occurrence of a manifestation. An import of 5 (manifestation absolutely must be explained by one of the final diagnoses) signifies that the baseline occurrence of the manifestation is zero, or extremely small. Similarly, an import of 1 (manifestation is usually unimportant, occurs commonly in normal persons, and is easily disregarded) signifies that the baseline occurrence of that manifestation is relatively high; that is, the probability of a disease that can directly account for the manifestation is increased only slightly when compared to a disease for which no frequency number is explicitly encoded in the Internist-I knowledge base.

Although it would not have been difficult to estimate baseline probabilities for each manifestation de novo (and there are many advantages to doing so), we decided to use an import-to-manifestation baseline mapping in KBSimulator to explore whether that mapping would yield a knowledge base that could be used for teaching. From Table 1, we see that a frequency of 1 corresponds to a $P(Mx|Dx)$ of 0.025. The 'baseline occurrence' of any manifestation, then, should generally be smaller than this number (if it was higher, then a frequency number should have been encoded). For those manifestations with an import of 1, we set the baseline occurrence of that manifestation to 0.02, and create an exponentially decreasing function to calculate the baseline occurrence of manifestations with higher import numbers. The resultant mapping of imports to baseline occurrences is shown in Table 3, and is described by the equation:

$$P(Mx_{\text{baseline}}) = 0.02 * 2^{-(\text{Import}-1)}$$

TABLE 3

Mapping import to baseline manifestation occurrence

Import	$P(Mx_{\text{baseline}})$
1	0.02
2	0.01
3	0.005
4	0.0025
5	0.00125

Using this technique, we are able to fill in the missing probabilities in Table 2, to yield a knowledge base that contains the a priori probability of 31 cardiovascular diseases, and a $P(Mx|Dx)$ for the occurrence of each of 350 manifestations in each of the 31 diseases. Although numerous assumptions were made to derive this table, one of our objectives was to determine whether such a probabilistic conversion would yield a knowledge base adequate for teaching purposes.

4.2. Grouping manifestations

Attempting to simulate patients from a probabilistic knowledge base that assumes conditional independence among manifestations often results in the generation of unlikely or nonsensical patient cases [6]. An example of this would be the disease MITRAL REGURGITATION and the associated manifestations P-WAVE NOTCHED and ATRIAL FIBRILLATION. Since these manifestations are both present with significant probability in the disease MITRAL REGURGITATION, it would be possible in an unmodified Bayesian system to generate a patient who exhibited both of these manifestations. However, since the presence of atrial fibrillation precludes the possibility of seeing a p-wave on the electrocardiogram (EKG), a patient who had both these manifestations would present inconsistent findings. In addition, it would be desirable to minimize the presentation of manifestations that duplicate the same conceptual information. For example, it would not be useful to give the student the information that the patient had a decreased hematocrit when the system has already told him that the patient has a decreased hemoglobin, since the two observations generally convey the same information.

To attenuate the negative effects of the assumption of conditional independence among manifestations occurring in a disease, we grouped findings that are mutually exclusive or strongly dependent. Many of these relationships have been represented through the use of PROPERTIES in the Internist-I system [25]. In this project, we examined the list of manifestations and, with the assistance of the Internist-I PROPERTIES, formed groups of manifestations. Each group was classified as one of: Mutually Exclusive (e.g. MALE, FEMALE), Strongly Dependent (e.g. HEMATOCRIT, HEMOGLOBIN), or Both — i.e., a group that includes some manifestations that are mutually exclusive and some that are strongly dependent (e.g. TACHYCARDIA, EKG-TACHYCARDIA; BRADYCARDIA, and EKG-BRADYCARDIA).

As we shall see later, this group type is used to determine the probability of including a manifestation from that group when generating a patient case.

5. Patient generation

Patient cases that are generated from a knowledge base should be both realistic and useful for teaching. By pruning the manifestations in the manner described in Section 4, we selected those that would be useful to teach. By grouping manifestations, we provided enough structure in the knowledge base to avoid the generation of nonsensical patients. In this section, we describe how patient cases are generated (Fig. 2).

First, a disease is selected at random from the set of diseases about which the system wishes to teach the student. The system examines the list of manifestations associated with the selected disease, and determines the presence of each manifestation or manifestation group by generating a random number between 0 and 1 and comparing the generated number with the $P(Mx|Dx)$ for that manifestation or the $P(Gp|Dx)$ for that group of manifestations. For example, if $P(Mx|Dx)$ is 0.2, and the randomly generated number is equal to or below 0.2, that manifestation will be included in the patient case; if the generated number is above 0.2, the manifestation will not be included.

For each group of manifestations, a $P(Gp|Dx)$ (the probability of generating any of the manifestations in that group) is calculated. For Mutually Exclusive groups (groups that contain only mutually exclusive manifestations), the individual probabilities for each manifestation are summed (up to a maximum of 1.0) to form this 'group probability'. For Strongly Dependent groups, the highest probability number among all the manifestations is taken as the probability for the group. For groups that are classified as Both (contain both mutually exclusive and strongly dependent manifestations), the resulting probability for the overall group should be between the probability that would have been obtained if the groups contained only manifestations that were mutually exclusive or highly dependent. The following heuristic formula has the property that it results in a number that is between the probability obtained for the Mutually Exclusive and Strongly Dependent groups, and is used to determine the overall probability that should be assigned for groups classified as Both:

$$P(Gp|Dx) = 1 - [(1 - P(M_1|Dx)) * (1 - P(M_2|Dx)) * \ldots * (1 - P(M_n|Dx))] = 1 - \prod_i^n P(\overline{M_i}|Dx)$$

where $Gp = \{M_1, M_2, \ldots, M_n\}$.

An argument could be made for the use of a group of mutually exclusive subgroups of strongly dependent manifestations, and for the use of a combination of the techniques used for the Mutually Exclusive and Strongly Dependent groups to determine the overall probability for groups classified as Both. However, the use of the preceding heuristic formula led to reasonable behavior of the system and was adopted for simplicity.

If the randomization procedure determines that a group is present, then one of the manifestations in the group is chosen by another random draw. The probability that any particular manifestation will be selected from a chosen group is simply proportioned according to the $P(Mx|Dx)$ for that manifestation. Thus, at most, one manifestation in a group of mutually exclusive and/or strongly dependent manifestations can be present in a generated patient case.

Whenever a manifestation is determined to be present, it is added to the current generated patient case. Each patient case thus represents a list of manifestations that are present — the explicit *absence* of a manifestation is not represented. Up to ten such cases are generated at a time. After this case-generation phase, the program moves onto the patient-case-presentation and tutorial interaction phase.

6. Patient-case presentation and tutorial interaction

Figs. 3, 4, and 5 show annotated output from the system. Patient cases are presented to the student in a standard case-presentation fashion (Fig. 3).

The age and sex of the patient are presented first, followed by the patient's chief complaint and symptoms. History, physical findings, and laboratory and procedure results may also be presented. As each manifestation is presented to the student, KBSimulator uses Bayes' rule to recalculate the probability of the patient's diagnosis (see [13] and [36] for illustrations of the use of this approach). When the probability of a diagnosis exceeds a specified preset threshold, the system halts the presentation of further manifestations, as the student, we presume, has been provided with enough information to make the diagnosis.

At this point, the student has the option of entering a diagnosis (`Guess Dx`), asking for more information about his patient (`AnotherMx`), asking the system to indicate the manifestation that provides the strongest evident for the diagnosis (`Hint`), querying the knowledge base for the differential diagnoses of any manifestation present in this patient (`MxDiff`), moving on to the next case (`NextCase`), or redisplaying all the patient manifestations presented so far (`Redisplay`).

The system thus engages the student in a *mixed-initiative dialogue*, allowing the student to offer a diagnosis or to obtain further information of various kinds. For example, if the student were to ask for another manifestation, for the differential diagnosis of a manifestation, and then for a hint, the dialogue shown in Fig. 4 would ensue.

If the student, at some point, selects the correct diagnosis, KBSimulator reviews the case by indicating the manifestations that were the most important for the diagnosis. KBSimulator then presents the rest of the case to the student, showing

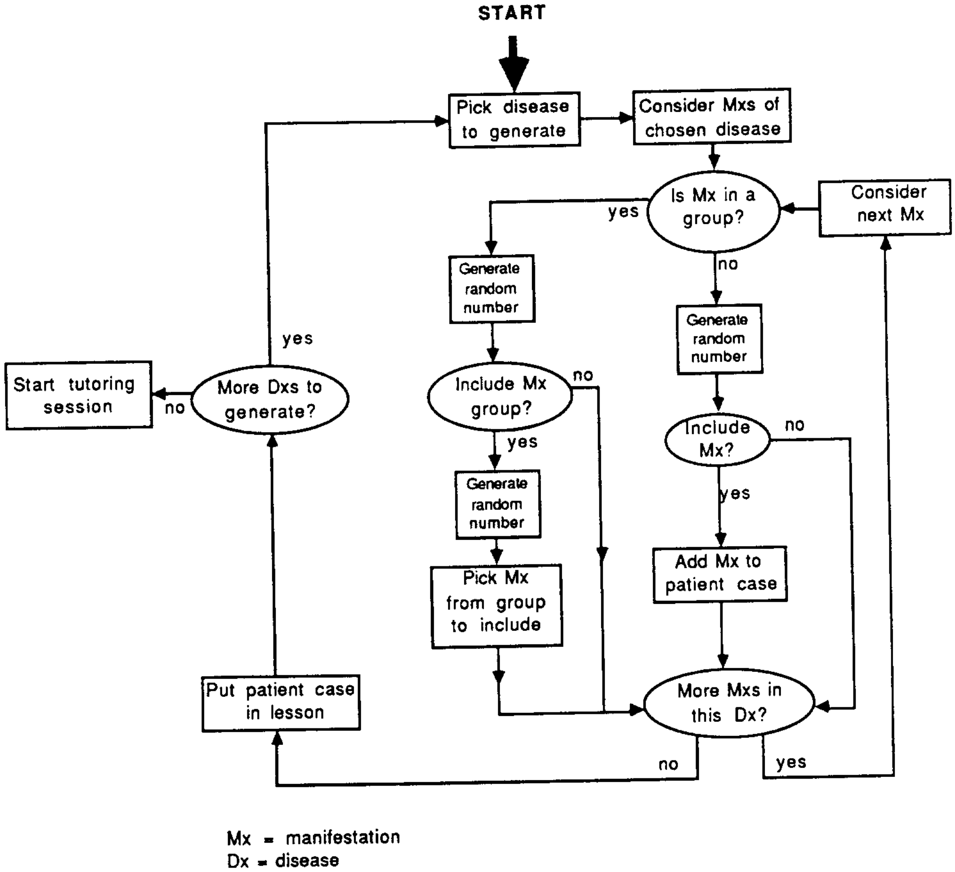

Fig. 2. Flow chart illustrating the process of patient-case generation from the knowledge base.

the student the remaining results of the history, physical examination, laboratory values, and the results of special procedures and studies. The student is allowed a final opportunity to explore the knowledge base prior to continuing on to the next case (Fig. 5).

```
This is a patient of SEX MALE, and of AGE 16 TO 25,
who has a chief complaint of
73: DYSPNEA AT REST.
History and physical exam show:
    52: CHEST TRAUMA REMOTE HX
   223: HEART SOUND <S> P2 INCREASED
   259: LEG <S> EDEMA BILATERAL SLIGHT OR MODERATE
   295: PRESSURE ARTERIAL SYSTOLIC 90 TO 110
   301: PRESSURE VENOUS INCREASED ON INSPECTION
   302: PRESSURE VENOUS KUSSMAUL SIGN POSITIVE
{at this point the program pauses because it has calculated a probability
of greater than the specified threshold (0.90) for the disease}
(G)uessDx, (A)notherMx, (H)int, (M)xDiff, (N)extCase, (R)edisplay:
```

Fig. 3. Initial case presentation (comments are enclosed in curly brackets).

Fig. 6 illustrates, in flow-chart form, the process of patient-case presentation used in KBSimulator. The program selects one of the previously generated patient cases and presents this case in a case-presentation fashion until the specified threshold for diagnosis is exceeded. The program then engages the student in a mixed-initiative dialogue, where the student is able to obtain further information on her own initiative. As we shall discuss in the next section, the program scores the student on each case presented. If the student selects the correct diagnosis, or if all the manifestations have been presented to the student and she is still unable to guess the correct diagnosis, the system reviews the case, displays the final score, and allows the student to obtain the differential

```
(G)uessDx,(A)notherMx,(H)int,(M)xDiff,(N)extCase,(R)edisplay: a
{student types "a," asking for another manifestation in this patient case}
  315: PULSE PRESSURE NARROW
(G)uessDx,(A)notherMx,(H)int,(M)xDiff,(N)extCase,(R)edisplay: m 315
{student requests the differential diagnosis of Mx # 315}
 Mx: PULSE PRESSURE NARROW
 Freq   Dx#:  Diagnosis
 0.2    26: RHEUMATIC CARDITIS ACUTE
 0.2    25: SEPTIC SHOCK
 0.5    20: MYOCARDITIS ACUTE
 0.5    19: MYOCARDIAL INFARCTION ACUTE
 0.9    16: HYPOVOLEMIC SHOCK
 0.8    14: HEAT EXHAUSTION
 0.5    11: CONSTRICTIVE PERICARDITIS
 0.2    10: CARDIOMYOPATHY PRIMARY
 0.8     9: CARDIOGENIC SHOCK ACUTE
 0.98    8: CARDIAC TAMPONADE
 0.2     7: CARDIAC FAILURE LEFT CHRONIC CONGESTIVE
 0.2     4: AORTIC VALVULAR STENOSIS
{differential diagnosis for Mx number 315 is printed out, with the
probability of finding that manifestation (PULSE PRESSURE NARROW) in each
of the diagnoses}
(G)uessDx,(A)notherMx,(H)int,(M)xDiff,(N)extCase,(R)edisplay: H
{user requests a hint}
This is a rare diagnosis, but the manifestation 52 CHEST TRAUMA REMOTE HX
is the most helpful in making the diagnosis.
```

Fig. 4. Dialogue between the program and student in the continuation of the case initially presented in Fig. 3 (user input is underlined, comments are enclosed in curly brackets).

diagnosis of any of the manifestations in the case before proceeding.

7. Student modeling

The system monitors a student's performance by keeping a score for each diagnosis that the student attempts. The student starts out with a score of 100 for each patient case. If the student is able to determine the correct diagnosis without any assistance, his score for that case is 100. For each hint, manifestation differential, and additional manifestation that the student requests, a penalty of 10 points is deducted. If he makes an erroneous guess of the diagnosis, his score is reduced by 50%. These are arbitrary prototypical numbers that can be adjusted as experience with the system increases. With further work, it will be possible to tailor the precise number of points deducted to the type of information requested, and to the extent of the error in diagnosis.

The student is given up to ten patient cases during each 'round'. After each round, the diseases for which the student diagnosed a case successfully with a score of 90 or greater are removed from subsequent simulation. As the interaction proceeds, only those diseases that the student had difficulty diagnosing are generated. In this way, the system uses the student's score to focus attention on those diseases with which the student has had the most difficulty. After each round, the student is told his average score for that round; at the conclusion of the entire interaction, the student is given a summary of the number of patient cases he has diagnosed, and the average score for the entire session. As the student learns the manifestation–disease relations, he can be expected to improve his performance and score, and therefore to obtain increasing positive feedback from the system.

```
(G)uessDx,(A)notherMx,(H)int,(M)xDiff,(N)extCase,(R)edisplay: G 11
{the student guesses Dx #11}
Dx: CONSTRICTIVE PERICARDITIS
Good!! You got it right!!
This is a rare diagnosis, but the manifestation 52 CHEST TRAUMA REMOTE HX and the
manifestation 302 PRESSURE VENOUS KUSSMAUL SIGN POSITIVE are the most helpful
in making the diagnosis.
{The system critiques this case by giving the most important manifestations that contributed to
the probability of this diagnosis, and then presents the rest of the manifestations in this patient
case}
This patient also had the following other findings:
  103: EKG T WAVE <S> INVERTED
  244: HEART XRAY PERICARDIAL CALCIFICATION
  241: HEART XRAY LEFT VENTRICLE ENLARGED
  303: PROTEINURIA
  262: LUNG <S> FORCED VITAL CAPACITY DECREASED
  128: HEART CATHETERIZATION LEFT ATRIUM MEAN PRESSURE INCREASED
  137: HEART CATH LEFT VENTRICLE END-DIASTOLIC PRESSURE INCREASED
  148: HEART CATHETERIZATION RIGHT VENTRICLE DIASTOLIC DIP
  140: HEART CATH PULMONARY ARTERY WEDGE PRESSURE INCREASED
  216: HEART OUTPUT DECREASED
  298: PRESSURE VENOUS CENTRAL GTR THAN 10
  146: HEART CATHETERIZATION RIGHT ATRIUM MEAN PRESSURE INCREASED
(M)xDiff,(N)extCase,(E)ndLesson: N
{The system gives the student another opportunity to find out the differential diagnosis of any
of the manifestations presented, before going on to the next case}
```

Fig. 5. Review and presentation of remaining patient case (user input is underlined, comments are enclosed in curly brackets).

8. Preliminary experience

KBSimulator has been implemented on a DEC-2060 computer and is accessible to all users of the SUMEX-AIM system at Standord University. The tutoring program itself was written in Standard Pascal, and comprises approximately 800 lines of code.

KBSimulator was built primarily to explore the use of ICAI techniques that would utilize an existing knowledge base to generate realistic patient cases that would be useful for teaching. In a preliminary evaluation of KBSimulator, four people — one medical student in his clinical years, two board-certified internists, and a pediatrician — interacted with the system.

All participants reported learning something useful from their interaction. A typical response

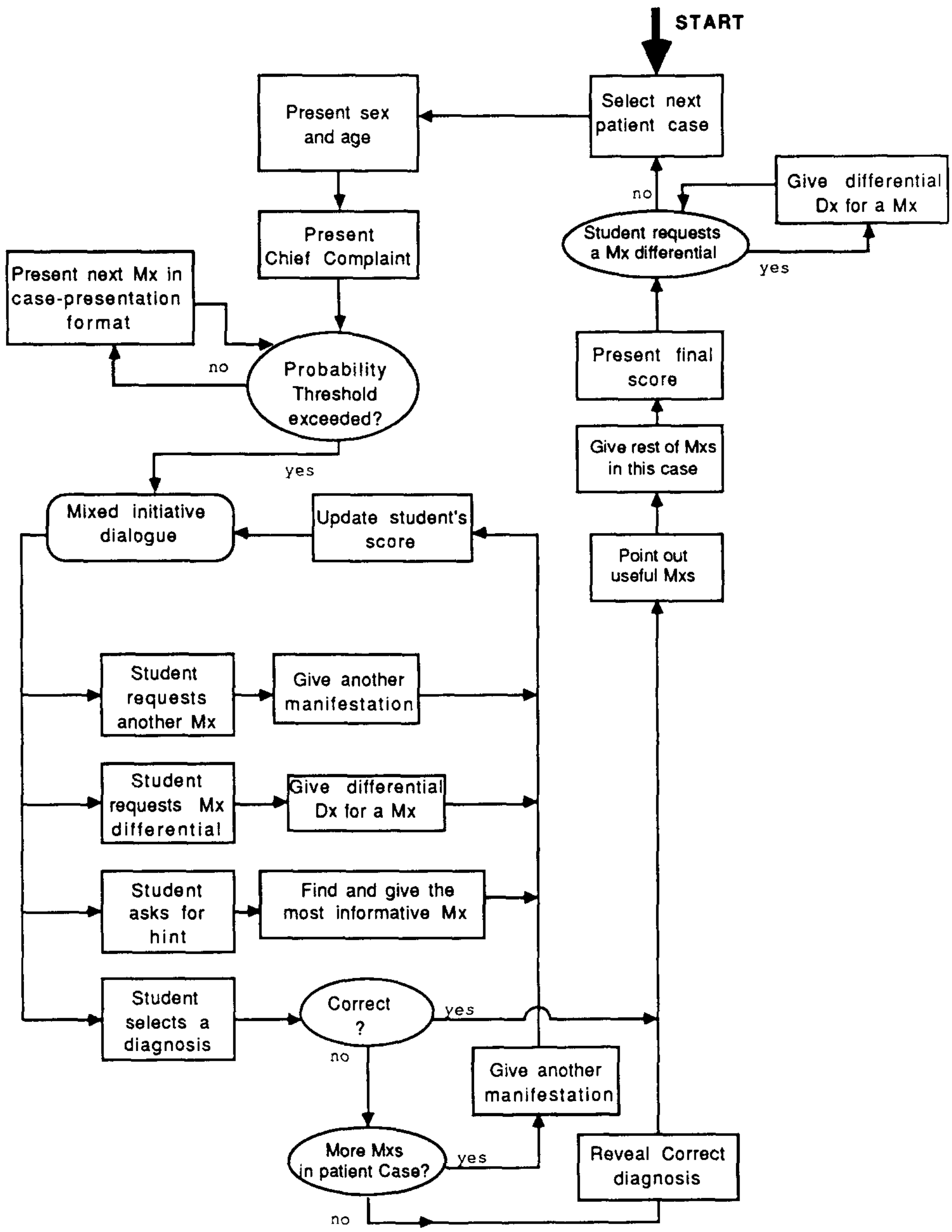

Fig. 6. Flow chart illustrating the process of patient-case presentation and student interaction.

was: 'KBSimulator reinforced or taught me the relationship of different heart murmurs in the diagnosis of cardiovascular disease.' Preliminary responses indicated that users enjoyed using the system, thought the simulated patients were realistic, felt that the interaction was a useful learning experience, and believed that this technique should be extended to other medical areas other than cardiology.

Most of the criticisms of KBSimulator related to the awkwardness of the user interface and the lack of text-recognition capability. In addition to improving the interface, suggestions for improvements were for the addition of 'red herring' manifestations, graphical displays of how manifestations contributed to the diagnosis, rewarding of partial success, and addition of the capability to explain the causal mechanisms that relate manifestation and disease.

Although this preliminary feedback is anecdotal, early experience suggests significant potential in the use of this approach for the construction of tutoring systems.

9. Discussion and conclusion

The main components of an ICAI system are [3]:

(1) A knowledge base (the knowledge that the system tries to impart to the student).

(2) Problem solving expertise.

(3) The student model, indicating what the student does and does not know.

(4) Tutoring strategies, which specify how the system presents material to the student.

KBSimulator has all the components of an ICAI system (at varying levels of sophistication) and, to our knowledge, it is the first system that makes use of a knowledge base *both* to generate simulated patients *and* to tutor a student intelligently. The methods used and described in this paper are by no means confined to this particular knowledge base or domain; they could be applied easily to any knowledge base that is encoded in a probabilistic format.

9.1. Patient-case generation from a knowledge base

By pruning the knowledge base of nonspecific manifestations, we were able to improve the ability to produce didactically useful patient cases. Through the use of manifestation grouping, we were able to encode some of the strong dependencies among manifestations, eliminating the generation of nonsensical patients and patients with redundant manifestations. The use of a *probabilistic causal network* representation (sometimes called a Bayesian belief network representation) would allow for the encoding of all known dependencies among manifestations without requiring that the entire joint probability space be specified [11,32]. Although the possible exponential time complexity of calculation needed to make inferences with such systems remains a problematic issue [7,12], this knowledge-representation technique holds much promise. The current lack of a large knowledge base in a causal probabilistic format precludes the adaptation of this method of knowledge representation for teaching at this time.

9.2. Student modeling

The use of a knowledge base to generate patient cases flexibly allows for the tailoring of the generated cases to the learning needs of the student. KBSimulator makes use of a rudimentary student model that contains simply the disease-specific scores that the student achieves when diagnosing patient cases. By using this model, KBSimulator is able to focus patient-case generation precisely on those diseases for which the student does not achieve a threshold score. By extending the model of the student to incorporate a measure of the student's understanding of each manifestation, patient-case generation could be tailored to generate cases that have the specific manifestations that the student does not understand.

9.3. Mixed-initiative dialogue

The use of a mixed-initiative dialogue allows the student to obtain specific information according to her needs and to explore the knowledge base in the context of diagnosing a specific patient case. This feature allows the individual student to obtain information that specifically addresses the gaps in her own fund of knowledge. Future work could be directed toward improving the nature of this mixed-initiative dialogue, by making better use of the 'expert' aspect of KBSimulator to tutor the specific misconceptions of the student. For example, the system could be enhanced such that, when the student makes an erroneous diagnosis, the system calculates the probability of that diagnosis to determine the extent of the error and asks the student which manifestations led her to make that particular diagnosis. The student could then be tutored specifically on the differences between the guessed disease and the correct one.

9.4. *Knowledge-base augmentation*

Much future work remains to be done in providing more than a numerical description of how manifestations and diseases are related. If we included a textual explanation of these relationships, the system would be capable of explaining the causal mechanisms involved. By limiting the textual explanations to only those manifestation-disease pairs that are of significant clinical importance, we could limit the amount of knowledge-acquisition work necessary while still retaining much of the benefit of this addition.

9.5. *Internist-I and probabilistic knowledge bases*

Our research into the probabilistic meaning of the numerical quantities used in the Internist-I knowledge base, and our demonstration of the ability to convert part of the Internist-I knowledge base into a probabilistic form that can be used for case-based tutoring, has several important implications. Because of the time-consuming work needed to create a knowledge base such as that used by Internist-I, other researchers have been interested in adapting the Internist-I knowledge base for various applications [16,30]. Conversion of the Internist-I knowledge base into a probabilistic format allows this knowledge to be encoded in the standard language of probability that is widely understood, so researchers can adapt such a knowledge base to their own applications. The use of a probabilistic methodology also allows one to incorporate statistical data into portions of the knowledge base, and to cross-check numbers in the knowledge base with statistical data as they become available.

9.6. *Future*

The goal of the research described in this paper was to explore the feasibility of using a medical knowledge base *both* to generate patient cases *and* to use ICAI techniques for flexible intelligent tutoring. Although we have succeeded in showing the feasibility of the methodology illustrated in this paper, much further potential exists in this research direction. By making use of a *causal network* representation, by expanding the student model, by improving the mixed-initiative dialogue to respond to the specific misconceptions of the student, by augmenting the knowledge base with 'support' knowledge, it will be possible to extend the work described in this paper to build a powerful and flexible learning system to augment the education of the budding physician.

Acknowledgments

We are indebted to Randy Miller for facilitating access to use the Internist-I knowledge base. Our collaborating cardiologist, William Hancock, helped with the development of the KBSimulator knowledge base. Lyn Dupre provided many useful editorial comments on earlier drafts of this paper. This work has been supported by grant LM-07033 from the National Library of Medicine. Computer facilities were provided by the SUMEX-AIM resource under NIH grant RR-00785.

References

[1] I.D. Adams, M. Chan, F.T. de Dombal et al., Medical practice: contemporary themes: computer aided diagnosis of acute abdominal pain: a multicentre study, Br. Med. J. 293 (1986) 800–804.

[2] G.O. Barnett, J.J. Cimino, J.A. Jupp and E.P. Hoffer, DXplain: an evolving diagnostic decision-support system, J. Am. Med. Assoc. 258 (1987) 67–74.

[3] A. Barr and E.A. Feigenbaum, Applications-oriented AI research: education, *in*: The Handbook of Artificial Intelligence, pp. 225–294 (William Kaufmann, Los Altos, CA, 1982).

[4] M.S. Blois, M.S. Tuttle and D.D. Sherertz, RECONSIDER: a program for generating differential diagnoses, *in*: Proceedings of the Fifth Annual Symposium on Computer Applications in Medical Care, pp. 263–266 (IEEE, New York, 1981).

[5] J.R. Carbonell, AI in CAI: an artificial-intelligence approach to computer-assisted instruction, IEEE Trans. Man-Mach. Sys., MMS-11 (1970) 190–202.

[6] H.L. Chin and G.F. Cooper, The role of causal knowledge in knowledge-based patient simulation, *in*: Proceedings of the Symposium on Computer Applications in Medical Care, pp. 460–466 (IEEE, New York, 1987).

[7] H.L. Chin and G.F. Cooper, Stochastic simulation of causal Bayesian belief models, *in*: Proceedings of the Third Workshop on Uncertainty in Artificial Intelligence,

pp. 106–113 (AAAI, 1987). [To appear as a chapter in Uncertainty in Artificial Intelligence, Vol. 3 (L.N. Kanal, T.S. Levitt and J.F. Lemmer, eds.), Elsevier/North-Holland, Amsterdam].

[8] W.J. Clancey, E.H. Shortliffe and B.G. Buchanan, Intelligent computer-aided instruction for medical diagnosis, *in*: Proceedings of the Third Annual Symposium on Computer Applications in Medical Care, pp. 175–183 (IEEE, New York, 1979).

[9] W.J. Clancey, Tutoring rules for guiding a case method dialogue, Int. J. Man-Mach. Stud. 11 (1979) 25–49.

[10] W.J. Clancey, From GUIDON to NEOMYCIN and HERACLES in twenty short lessons: ONR final report 1979–1985, AI Mag. August (1986) 40–60.

[11] G.F. Cooper, Expert Systems Based on Belief Networks. Technical Report KSL-87-51 (Medical Computer Science Group, Stanford University, Stanford, CA, 1987).

[12] G.F. Cooper, Probabilistic Inference Using Belief Networks is NP-Hard. Technical Report KSL-87-27 (Medical Computer Science Group, Stanford University, Stanford, CA, 1987).

[13] F.T. de Dombal, J.C. Horrocks, J.R. Staniland and P.W. Gill, Simulation of clinical diagnosis, Br. Med. J. 2 (1971) 575–577.

[14] F.T. de Dombal, D.J. Leaper, J.R. Staniland, A.P. McCann and J.C. Horrocks, Computer-aided diagnosis of acute abdominal pain, Br. Med. J. 2 (1972) 9–13.

[15] B.B. Farquhar, E.P. Hoffer and G.O. Barnett, Patient simulations in clinical education, *in*: Information Technology in Health Science Education, ed. E.C. DeLand, pp. 397–422 (Plenum Press, New York, 1978).

[16] R.N. Goldberg and S.M. Weiss, An experimental transformation of a large expert knowledge base, J. Med. Syst. 6 (1982) 41–52.

[17] W.G. Harless, G.G. Drennon, J.J. Marxer, J.A. Root and G.E. Miller, CASE: a computer-aided simulation of the clinical encounter, J. Med. Educ. 46 (1971) 443–448.

[18] D.E. Heckerman and R.A. Miller, Towards a better understanding of INTERNIST-I knowledge base, *in*: MEDINFO 86, eds. R. Salamon, B. Blum and M. Jorgensen, pp. 22–26 (1986).

[19] D.E. Heckerman, Probabilistic interpretations for MYCIN's certainty factors, *in*: Uncertainty in Artificial Intelligence, eds. L.N. Kanal and J.F. Lemmer, pp. 167–196 (Elsevier/North-Holland, Amsterdam, 1986).

[20] E.P. Hoffer, G.O. Barnett, B.B. Farquhar and P. Prather, *in*: Computer-Aided Instruction in Medicine, pp. 103–118 (Yearbook Medical Publishers, Chicago, IL, 1975).

[21] E.P. Hoffer, G.O. Barnett, K.T. Famiglietti and J.A. Laing, Jr., Computer-aided instruction in medicine: 16 years of MGH experience, *in*: MEDINFO 86, eds. R. Salamon, B. Blum and M. Jorgensen, pp. 920–922 (1986).

[22] E.J. Horvitz and D.E. Heckerman, The inconsistent use of measures of certainty in artificial intelligence research, *in*: Uncertainty in Artificial Intelligence, eds. L.N. Kanal and J.F. Lemmers, pp. 137–151 (Elsevier/North-Holland, Amsterdam, 1986).

[23] L.J. Kozak and M. Moien (National Center for Health Statistics), Detailed diagnoses and surgical procedures for patients discharged from short-stay hospitals, United States, 1983, *in*: Vital Health Statistics, Ser. 13, No. 82, DHHS Publication No. (PHS)85-1743 (US Government Printing Office, Washington, DC, 1985).

[24] B. Kuipers and J.P. Kassirer, Causal reasoning in medicine: analysis of a protocol, Cognitive Sci. 8 (1984) 363–385.

[25] F.E. Masarie, Jr., R.A. Miller and J.D. Myers, Internist-I properties: representing common sense and good medical practice in a computerized medical knowledge base, Comput. Biomed. Res. 18 (1985) 458–479.

[26] R.A. Miller, J.E. Pople, Jr. and J.D. Myers, INTERNIST-I, an experimental computer-based diagnostic consultant for general internal medicine, New Engl. J. Med. 307 (1982) 468–476.

[27] R.A. Miller, A computer-based patient case simulator, Clin. Res. 32 (1984) 651A (abstract).

[28] R.A. Miller, M.A. McNeil, S.M. Challinor, F.E. Masarie and J.D. Myers, The INTERNIST-I/QUICK MEDICAL REFERENCE project — Status report, West. J. Med. 145 (1986) 816–822.

[29] S. Muller (Chair), Physicians for the 21st century — Report of the project panel on general professional education of the physician and college preparation for medicine, J. Med. Educ. 59 (11, Part 2) (1984) 1–208.

[30] L.T. Nguyen, Transferability of a Medical Knowledge Base: A Case Study Between INTERNIST-I and HELP (PhD thesis) (University of Utah, 1985).

[31] R.C. Parker and R.A. Miller, Using causal knowledge to create simulated patient cases: the CPCS project as an extension of INTERNIST-I, *in*: Proceedings of the Symposium on Computer Applications in Medical Care, pp. 473–480 (IEEE, New York, 1987).

[32] J. Pearl, Fusion, propagation and structuring in belief networks, Artif. Intell. 29 (1986) 241–288.

[33] T.A. Pryor, R.M. Gardner, P.D. Clayton and H.R. Warner, The HELP system, J. Med. Syst. 7 (1983) 87–102.

[34] E.H. Shortliffe, Computer-Based Medical Consultations: MYCIN (Elsevier/North-Holland, Amsterdam - New York, 1976).

[35] D. Sleeman and J.S. Brown, Intelligent Tutoring Systems (Academic Press, London, 1982).

[36] H.R., Warner, A.F. Toronto, L.G. Veasey and R. Stephenson, A mathematical approach to medical diagnosis: application to congenital heart disease, J. Am. Med. Assoc. 177 (1961) 177–183.