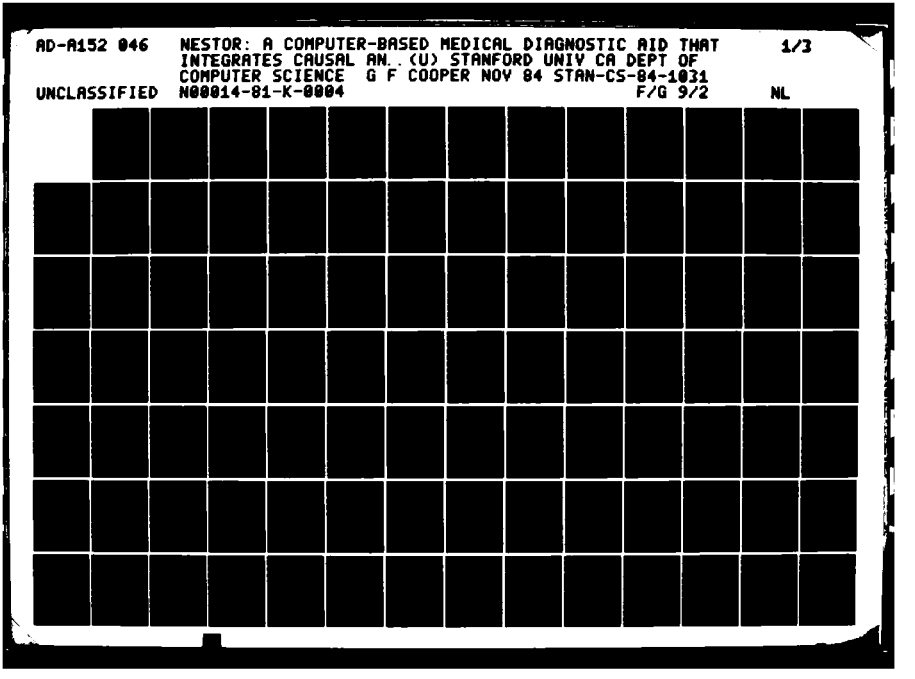
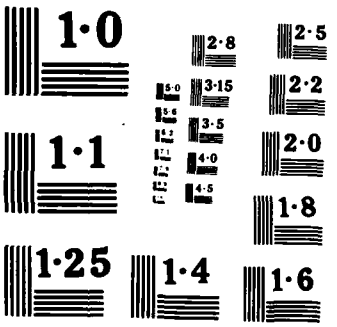

AD-A152 046 NESTOR: A COMPUTER-BASED MEDICAL DIAGNOSTIC AID THAT INTEGRATES CAUSAL AN..(U) STANFORD UNIV CA DEPT OF COMPUTER SCIENCE G F COOPER NOV 84 STAN-CS-84-1031
1/3
UNCLASSIFIED N00014-81-K-0004 F/G 9/2 NL


1·0
1·1
1·25
2·8
3·15
3·5
4·0
4·5
5·0
1·4
2·5
2·2
2·0
1·8
1·6

November 1984

Report No. STAN-CS-84-1031
Also numbered: HPP-84-48

(2)

NESTOR:
A Computer-Based Medical Diagnostic Aid that Integrates Causal and Probabilistic Knowledge

by

Gregory Floyd Cooper

Departments of Medicine and Computer Science

Stanford University
Stanford, CA 94305

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1 REPORT NUMBER	2. GOVT ACCESSION NO: AD-A152 046
	3. RECIPIENT'S CATALOG NUMBER
4 TITLE (and Subtitle): NESTOR: A Computer-Based Medical Diagnostic Aid that Integrates Causal and Probabilistic Knowledge	5. TYPE OF REPORT & PERIOD COVERED: Technical
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Gregory F. Cooper	8. CONTRACT OR GRANT NUMBER(s): ONR N00014-81-K-0004
9 PERFORMING ORGANIZATION NAME AND ADDRESS: Department of Computer Science, Stanford University, Stanford, CA 94305 USA	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11 CONTROLLING OFFICE NAME AND ADDRESS: Personnel and Training Research Programs, Office of Naval Research (Code 458), Arlington, VA 22217	12. REPORT DATE: November 1984; 13. NO. OF PAGES: 251
	15. SECURITY CLASS. (of this report): Unclassified
14 MONITORING AGENCY NAME & ADDRESS (if diff. from Controlling Office): ONR Representative-Mr. Robin Simpson, Durand Aeronautics Building, Room 165, Stanford University, Stanford, CA 94305	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16 DISTRIBUTION STATEMENT (of this report)

Approved for public release; distribution unlimited

17 DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from report)

18 SUPPLEMENTARY NOTES

19 KEY WORDS (Continue on reverse side if necessary and identify by block number)

Artificial intelligence	Critiquing
Expert systems	Explanation
Medical applications	Bayesian scoring
Branch and bound search	Causal reasoning

20 ABSTRACT (Continue on reverse side if necessary and identify by block number)

In order to address some existing problems in computer-aided medical decision making, a computer program called NESTOR has been developed to aid physicians in determining the most likely diagnostic hypothesis to account for a set of patient findings. The domain of hypercalcemic disorders is used to test solution methods that should be applicable to other medical areas.

A key design philosophy underlying NESTOR is that the physician should have control of the computer interaction to determine what is done and when. In order to provide

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

19. KEY WORDS (Continued)

20 ABSTRACT (Continued)

such a controllable, interactive aid, specific technical tasks had to be addressed. The unifying philosophy in addressing them is the use of knowledge-based methods within a formal probability theory framework. The tasks are as follows:

1. Scoring Hypotheses: The likelihood of an hypothesis is determined by using formal probabilistic reasoning with heuristic knowledge introduced explicitly as assumptions. During the scoring process causal knowledge is used in guiding the application of relatively sparse probabilistic knowledge. The scoring method emphasizes bounding the probability of an hypothesis rather than calculating it exactly.

2. Searching Hypotheses: A branch and bound search technique is used to search among diagnostic hypotheses for the most probable one. This technique is able to locate the most probable hypothesis without exploring the entire hypothesis space, which is particularly useful in diagnosing complex, multidisease cases where the space may be very large.

3. Explanation: NESTOR is able to critique and compare hypotheses which are generated by the system, volunteered by the user, or both. Critiquing a single hypothesis involves explaining the critical qualitative causal and quantitative probabilistic factors that affect its score. In addition, any two hypotheses can be comparatively critiqued.

A user interface module gives the physician control over when and how these tasks are used to aid in diagnosing the cause of a patient's condition.

This dissertation presents the problems that are addressed by each of the three tasks, and the details of the methods used to address them. In addition, the results of an evaluation of the hypothesis scoring and search techniques are presented and discussed.

NESTOR: A Computer-Based Medical Diagnostic Aid that Integrates Causal and Probabilistic Knowledge

A DISSERTATION

SUBMITTED TO THE COMMITTEE ON GRADUATE STUDIES

OF STANFORD UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

IN

MEDICAL INFORMATION SCIENCES

By

Gregory Floyd Cooper

November 1984

Abstract

In order to address some existing problems in computer-aided medical decision making, a computer program called NESTOR has been developed to aid physicians in determining the most likely diagnostic hypothesis to account for a set of patient findings. The domain of hypercalcemic disorders is used to test solution methods that should be applicable to other medical areas.

A key design philosophy underlying NESTOR is that the physician should have control of the computer interaction to determine what is done and when. In order to provide such a controllable, interactive aid, specific technical tasks had to be addressed. The unifying philosophy in addressing them is the use of knowledge-based methods within a formal probability theory framework. The tasks are as follows:

1. Scoring Hypotheses: The likelihood of an hypothesis is determined by using formal probabilistic reasoning with heuristic knowledge introduced explicitly as assumptions. During the scoring process causal knowledge is used in guiding the application of relatively sparse probabilistic knowledge. The scoring method emphasizes bounding the probability of an hypothesis rather than calculating it exactly.

2. Searching Hypotheses: A branch and bound search technique is used to search among diagnostic hypotheses for the most probable one. This technique is able to locate the most probable hypothesis without exploring the entire hypothesis space, which is particularly useful in diagnosing complex, multidisease cases where the space may be very large.

3. Explanation: NESTOR is able to critique and compare hypotheses which are generated by the system, volunteered by the user, or both. Critiquing a single hypothesis involves explaining the critical qualitative causal and quantitative probabilistic factors that affect its score. In addition, any two hypotheses can be comparatively critiqued.

A user interface module gives the physician control over when and how these tasks are used to aid in diagnosing the cause of a patient's condition.

This dissertation presents the problems that are addressed by each of the three tasks, and the details of the methods used to address them. In addition, the results of an evaluation of the hypothesis scoring and search techniques are presented and discussed.

Acknowledgements

The development of this dissertation has depended on the help of many people. I would like to especially thank Ted Shortliffe, my principal thesis advisor, for his guidance, support, and encouragement. I would also like to thank the other members of my reading committee: Bruce Buchanan, who generously aided me in starting my research at Stanford. Larry Crapo, who has devoted many hours to helping me construct NESTOR's knowledge-base, and Doug Lenat. In early research leading to the development of NESTOR, Casey Quayle was a patient source of great help in teaching me the details of INTERNIST's knowledge representation. In the early design phases of NESTOR, Randy Miller provided me with some very useful feedback.

There are a number of people who have influenced the direction of my career path leading to this research. I would like to especially acknowledge the influence of Lilla Burns, Sandy Mesel, David Meyers, Pete Szolovits, Lester Thomasson, and David Wirtschafter.

I want to thank some of my close friends, who have seen me through the ups and downs of work on this dissertation: Jim Brinkley, Bill Clancey, Claire Colley, Faramarz Keyvanfar, and Thomas Lengauer.

Finally, I am most grateful to my family and particularly my parents, Floyd and Jacqueline Cooper, for all they have given me, not just during these last few years, but throughout my life.

Funding for this research was provided in part by the Medical Scientist Training Program under NIH grant GM-07365, the Office of Naval Research under ONR contract N00014-81-K-0004, the National Library of Medicine under grant LM-03395, and a grant from the Henry J. Kaiser Family Foundation. The computation facilities were provided by SUMEX-AIM under grant RR-00785 from the Biotechnology Resources Program of the NIH.

Table of Contents

1. Introduction 1

1.1. The Motivation for Computer-aided Medical Decision Making 2
1.2. Where NESTOR Fits within Computer-aided Medical Decision Making 3
1.3. The Tasks Performed by NESTOR 5
1.4. NESTOR as a Medical Decision Support System 6
1.4.1. The Concept of a Medical Decision Support System 6
1.4.2. The Historical Progression Toward Medical Decision Support Systems 7
1.5. An Overview of NESTOR's Diagnostic Methods 8
1.5.1. Scoring an Hypothesis 9
1.5.1.1. Previous Approaches 9
1.5.1.2. NESTOR's Approach 15
1.5.2. Searching for the Most Probable Hypothesis 29
1.5.2.1. Previous Approaches 29
1.5.2.2. NESTOR's Approach 30
1.5.3. Explanation 31
1.5.3.1. Previous Approaches 31
1.5.3.2. NESTOR's Approach 31
1.6. The Domain of Application 33
1.7. Design Assumptions 34
1.8. Overview of Chapters 35

2. Examples **37**

3. Knowledge Representation **56**

3.1. Nodes 56
3.1.1. Disease Representation 59
3.1.2. Finding Representation and Entry 60
3.1.2.1. Examples of Entering Findings 60
3.1.2.2. The General Algorithm for Entering Findings 63
3.1.3. Knowledge Aquisition of Nodes 66
3.2. Links 66
3.2.1. The Representation of Links 66
3.2.1.1. A Two Level Hierarchical Model 66

3.2.1.2. Link Restrictions in NESTOR 67
3.2.1.3. A Link Viewed as a Probability Density Function 70
3.2.2. The Acquisition of Links 70
3.2.3. The Use of Links 74
3.2.3.1. Computing Conditional Probabilities 74
3.2.3.2. The Importance of Probabilistic Bounding in Representing Conditional Probabilities 75

4. Scoring a Diagnostic Hypothesis 78

4.1. The Scoring Metric 78
4.1.1. Bayes' Formula 79
4.1.2. NESTOR's Scoring Metric 81
4.1.2.1. The Causal Interpretation of the Scoring Metric 82
4.1.2.2. Bounding the Scoring Metric 83
4.2. The Scoring Algorithm: An Overview 84
4.2.1. The Nature of Causal Knowledge in NESTOR 84
4.2.2. Causal simulation 86
4.2.3. A Brief Example 86
4.3. The Scoring Algorithm: The Details 92
4.3.1. Step1: Create a Patient-Specific Causal Graph 92
4.3.2. Step2: Segment the Causal Graph into Levels 94
4.3.3. Step3: Compute the Score from the Segmented Causal Graph 95
4.3.3.1. Computing the Probability of Level $j+1$ Given Level 1 to Level j 96
4.3.3.2. Calculating P(H) 126
4.4. Related Work 128
4.4.1. A Description of the Features Being Compared 128
4.4.2. Comments on Each Program 130
4.4.3. A Comparison of NESTOR to Previous Research 135
4.4.4. Features Lacking in All the Programs 136
4.5. An Evaluation of NESTOR's Scoring Method 136
4.5.1. Methods 137
4.5.1.1. The Two Computer Programs being Compared 137
4.5.1.2. Case Classification 138
4.5.1.3. Case Generation 138
4.5.1.4. Case Evaluation 140
4.5.2. Results 141
4.5.2.1. Analysis of the Cases Generated by Random Sampling 143
4.5.2.2. Analysis of the Cases in which NESTOR and BAYES Differed 145
4.5.3. Discussion 152
4.6. Extensions 155

4.6.1. Caching Subtree Values 155
4.6.2. Handling Unreliable Findings 157
4.6.3. General Multivariable Links 158
4.6.4. Arbitrary Intermixing of Categorical and Probabilistic Knowledge 158
4.6.5. Nosological Representation 158
4.6.6. Generalizing the Temporal Representation of Links 159
4.6.7. Causal Feedback 161

5. Searching for the Most Probable Diagnostic Hypothesis 163

5.1. Background 163
5.2. The Precise Goal of NESTOR's Hypothesis Search Method 165
5.3. An Example of NESTOR's Hypothesis Search Method 165
5.4. A General Branch and Bound Algorithm for Determining the Most Probable Diagnostic Hypothesis 172
5.5. User Options to Direct Hypothesis Searching 172
5.5.1. Limiting the Amount of Search Time 172
5.5.2. Finding the N Most Likely Hypotheses 172
5.5.3. Using Successive Sets of Assumptions to Find the Most Probable Hypothesis 174
5.5.4. Inclusion of User-Specified Diseases in All Hypotheses 175
5.5.5. Exclusion of User-Specified Diseases from All Hypotheses 175
5.5.6. Limit the Number of Diseases in an Hypothesis 176
5.5.7. A User-specified Goal for the Precision of the Posterior Probability of the Most Probable Hypothesis 176
5.6. Related Work 177
5.6.1. A Description of the Features Being Compared 178
5.6.2. Comments on Each Program 179
5.7. An Evaluation of NESTOR's Hypothesis Search Method 183
5.7.1. Methods 185
5.7.2. Results 187
5.7.3. Discussion 188
5.7.3.1. Exhaustive Search 188
5.7.3.2. Pruned Search 190
5.7.3.3. Pruning with Heuristic Initialization and Best-first Ordering 190
5.7.3.4. The Extensibility of the Results 192
5.7.3.5. Summary 194
5.8. Extensions 195
5.8.1. Improved Branch and Bound Pruning 195
5.8.2. Searching in an Abstraction Space 197

6. Explanation **198**
6.1. The COMPARE Command 199
6.1.1. Step 1: Text Generation for Explaining Case-Specific Qualitative Causal Knowledge 200
6.1.2. Step 2: Comparing the Quantitative Likelihood of Two Hypotheses 203
6.2. The CRITIQUE Command 205
6.2.1. Calculating the Lower Bound of P(F) 207
6.2.2. Calculating the Upper Bound of P(F) 207
6.2.2.1. Method 1 208
6.2.2.2. Method 2 209
6.2.3. Goal-Oriented Tightening of Bounds on P(H | F) 210
6.2.4. The Source of Imprecision in P(H | F) 211
6.3. Related Work 212
6.4. Extensions 214
6.4.1. Tailored explanations 214
6.4.2. An Analysis of the Local Causal Factors Affecting the Relative Likelihood of Two Hypotheses 215
6.4.3. Extensions to the CRITIQUE Command 216
7. Summary and Conclusions **218**
7.1. Scoring a Diagnostic Hypothesis 218
7.1.1. Problems, Methods, and Results 218
7.1.2. Limitations and Future Research 223
7.2. Searching for the Most Probable Hypothesis 224
7.2.1. Problems, Methods, and Results 224
7.2.2. Limitations and Future Research 225
7.3. Explanation 225
7.3.1. Problems, Methods, and Results 225
7.3.2. Limitations and Future Research 226
7.4. NESTOR as a Medical Decision Support System 226

List of Tables

Table 3-1: The PDF Relating Serum Calcium Level to Level of Consciousness 71
Table 3-2: Partial PDF Relating Serum Calcium Level to Level of Consciousness 73
Table 3-3: Complete PDF Relating Serum Calcium Level to Level of Consciousness 74
Table 4-1: Convergence Probabilities If Two Causes Increase the Effect 101
Table 4-2: Convergence Probabilities If Two Causes Decrease the Effect 104
Table 4-3: Convergence Probabilities If One Cause Increases and Another Decreases the Effect 105
Table 4-4: Comparison of Diagnostic Programs that Use a Causal Model 129
Table 4-5: The Ratings of the Cases Generated by Random Sampling 143
Table 4-6: The Causes of Errors in the Cases Generated by Random Sampling 144
Table 4-7: The Comparative Ratings for the Cases in which NESTOR and BAYES Disagreed 145
Table 4-8: The Causes of NESTOR's Errors in the Cases in which NESTOR and BAYES Disagreed 147
Table 4-9: The Causes of BAYES' Errors in the Cases in which NESTOR and BAYES Disagreed 147
Table 5-1: A Comparison of the Hypothesis Search Methods of Selected Diagnostic Programs 177
Table 5-2: The Results of the Hypothesis Search Evaluation 188

List of Figures

Figure 1-1: A Decomposition of the Tasks in Medical Decision Making 4
Figure 1-2: The Subtask of Scoring an Hypothesis 10
Figure 1-3: Bayes' Formula 11
Figure 1-4: The Subtasks of Computing P(F | H) and P(H) in Scoring an Hypothesis 16
Figure 1-5: Using a Conditional Independence Assumption vs. Causal Knowledge in Calculating a Joint Conditional Probability 19
Figure 1-6: An Example of Causally Unstructured and Causally Structured Models 21
Figure 1-7: The Graph Used in a Probabilistic Causal Simulation 22
Figure 1-8: The Six Instantiations of the Graph in Figure 1-7 23
Figure 1-9: Joint Conditional Probabilities in a Causal Graph 26
Figure 1-10: Using Categorical and Probabilistic Causal Knowledge to Calculate a Joint Conditional Probability 28
Figure 3-1: A Partial Causal Graph of PHPT 57
Figure 3-2: An Example of a Finding Node 58
Figure 3-3: An Example of a Hierarchy of Node Values 58
Figure 3-4: An Example of a Disease Node 59
Figure 3-5: A Two Level Causal Model of Some of the Effects of Hypercalcemia 68
Figure 4-1: An Expanded Form of Bayes' Formula 79
Figure 4-2: Definition of Posterior Probability 80
Figure 4-3: The Adequacy of $P(F \mid H_i) \times P(H_i)$ for Rank Ordering 81
Figure 4-4: A Causal Graph 83
Figure 4-5: Causal Relationships Among Four Nodes in PHPT 85
Figure 4-6: Step 1: Causal Graph Generated for the Example 87
Figure 4-7: Step 2: Assigning Nodes to Levels in the Example 88
Figure 4-8: Conditional Probabilities Used in the Example 89
Figure 4-9: Step3: Calculating the Probability of Causal Graphs with Specific Value Instantiations for All Intermediate Variables 90
Figure 4-10: The Calculation of P(F | H) within the Context of Medical Decision Making 93
Figure 4-11: A Sample Segmentation of a Causal Graph 94

Figure 4-12: An Example of Causal Relationships Between Levels 96
Figure 4-13: An Example of the Merger of Convergent Links 97
Figure 4-14: An Example of the Merger of Divergent Links 97
Figure 4-15: The Merger of Links in a Convergent Group 98
Figure 4-16: Structure of the Convergent Merge Transformation 100
Figure 4-17: An Example Focusing on a Convergent Link Group 103
Figure 4-18: The Monotonicity and Conditional Probabilities of Convergent Links in the Example 104
Figure 4-19: An Example Focusing on a Divergent Link Group 113
Figure 4-20: Causal Knowledge for a Divergent Group, Example 1 114
Figure 4-21: Causal Knowledge for Divergent Group, Example 2 118
Figure 4-22: Summary of Local Joint Conditional Probability Calculations of an Example 122
Figure 4-23: The Causal Relationships Among Five Hypothetical Diseases 127
Figure 4-24: An Example of a Case Evaluation Sheet 142
Figure 4-25: Causally Unstructured vs. Structured Findings: Detecting Impossible Diagnoses 148
Figure 4-26: Causally Unstructured vs. Structured Findings: The Effect on Aggregating Findings 151
Figure 4-27: A Sample Causal Graph with Cached Values 156
Figure 4-28: Unfolding a Feedback Loop 162
Figure 5-1: An Example of an Hypothesis Tree Created During Hypothesis Search 168
Figure 5-2: A Branch and Bound Algorithm that Determines the Most Probable Diagnostic Hypothesis 173
Figure 5-3: The Subtask of Searching for the Most Probable Hypothesis 184
Figure 5-4: A Plot of the Relationship between the Number of Hypotheses Generated and the Score of the Best Hypothesis 189
Figure 6-1: The Subtasks of Critiquing and Comparing Diagnostic Hypotheses 199
Figure 6-2: An Example of the Comparison of Two Hypotheses 201
Figure 6-3: Using Detailed Causal Knowledge in an Explanation 203
Figure 6-4: The Statistic Used to Compare Two Hypotheses 204
Figure 6-5: An Example of Critiquing an Hypothesis 206
Figure 6-6: A Formula for Calculating the Posterior Probability of an Hypothesis 206
Figure 7-1: A Summary of the Tasks Performed by NESTOR 219

Chapter 1

Introduction

In order to address some existing problems in computer-aided medical decision making, a computer program called NESTOR[1] has been developed to aid physicians in determining the most likely diagnostic hypothesis to account for a set of patient findings. The domain of hypercalcemic disorders is used to test solution methods that should be applicable to other medical areas.

A key design philosophy underlying NESTOR is that the physician should have control of the computer interaction in order to determine what type of aid is given and when it is given. In order to provide such a controllable, interactive aid, specific technical tasks had to be addressed. The tasks involve scoring hypotheses, searching for a most probable hypothesis, and critiquing and comparing hypotheses. The unifying theme in addressing these tasks is the use of knowledge-based methods within a formal probability theory framework.

NESTOR uses both causal and probabilistic knowledge in determining the score (a probability) that is assigned to an hypothesis. This dissertation addresses the question of whether the additional use of causal knowledge improves the accuracy of scoring of hypotheses when compared to the use of probabilistic knowledge alone.

NESTOR is able to diagnose multiple disease hypotheses. Intuitively, in a typical clinical case it seems unnecessary to explicitly consider *all* possible disease combinations in order to locate the most likely diagnostic hypothesis. This dissertation addresses the question of whether this intuition is correct, thus making it is possible to develop a search

---

[1]The term NESTOR is taken from the name of a character in Greek mythology who provided well-respected advice and lived a very long life.

technique which must explore only a small portion of the entire hypothesis space in order to *guarantee* that the most likely hypothesis has been found.

NESTOR is able to critique and compare hypotheses which are generated by the system itself, volunteered by the user, or both. This dissertation develops a method of using both qualitative causal knowledge and quantitative probabilistic knowledge in critiquing and comparing hypotheses.

After a brief discussion of some general reasons why computer diagnostic aids are needed in medicine, a more detailed description of NESTOR's tasks and methods will be presented.

1.1. The Motivation for Computer-aided Medical Decision Making

In the past 25 years there has been a growing interest in using computers to aid in medical decision making [Ledley 59, Shortliffe 79]. A major driving force behind this interest is the realization that the quantity of clinically relevant medical knowledge is too great for one person to remember. The medical care system has adapted through the introduction of individual specialization to cope with this knowledge growth. Major disadvantages of specialized care are that it leads to fragmentation of medical care, and it is not readily available in some geographical locations. On the other hand, although general practitioners may be more accessible and able to offer more integrated care, their knowledge of medicine can often limit their effectiveness in dealing with specialized problems.

In addition to memory limitations, human problem solvers also have processing limitations. Research has illuminated some definite biases in decision making [Tversky 74], biases that often lead to errors in judgment. Detmer has shown that some of these same biases apply to physicians during diagnosis [Detmer 78]. In particular, studies have shown that physicians are not adept at correctly utilizing probabilistic information [Berwick 81]. Komaroff [Komaroff 79] summarizes well some of the human cognitive limitations discovered in the above research:

1. The mind fails to make proper quantitative assessments of immediate

experience: it fails to consider or comprehend the prior probability of an event, the importance of sample size, the laws of chance, and the realities of statistical phenomena such as regression toward the mean.

2. The mind fails to recall pertinent past experience: recent experience may be more accessible than more distant experience; an experience which seemed irrelevant or nonsensical at the time may be buried in the memory because there was no salient intellectual niche into which it could be fit; a hypothetical event may be difficult to imagine in quantitative terms, such as the degree of risk, if one has had little or no past experience which is relevant.

3. The mind places undue emphasis on its initial estimate of an event's likelihood in all subsequent estimates; similarly, the mind tends to estimate a confidence interval around an estimated mean value which is inappropriately narrow.

The potential power of computer programs lies precisely in those areas, such as probabilistic reasoning, where one encounters many of the limitations of human decision making. So, in addition to being a memory booster, the computer can potentially serve as a reasoning aid to the physician.

Although there are still many technical, legal, and sociological issues to be addressed in computer-aided medical decision making [Schwartz 70], a central tenet of this dissertation is that improving the individual physician's ability to intelligently utilize vast bodies of medical knowledge will improve the quality of medical care.

1.2. Where NESTOR Fits within Computer-aided Medical Decision Making

A computer-aided medical decision making program (CAMDM), called NESTOR, has been developed to aid in the diagnosis of the diseases that cause hypercalcemia. This domain provides a testing ground for ideas that are applicable to many other areas of medicine. Figure 1-1 shows a simplified decomposition of the tasks in medical decision making. The box encloses those tasks addressed by NESTOR. A major problem addressed by the program is to return the most likely hypotheses that account for a given set of

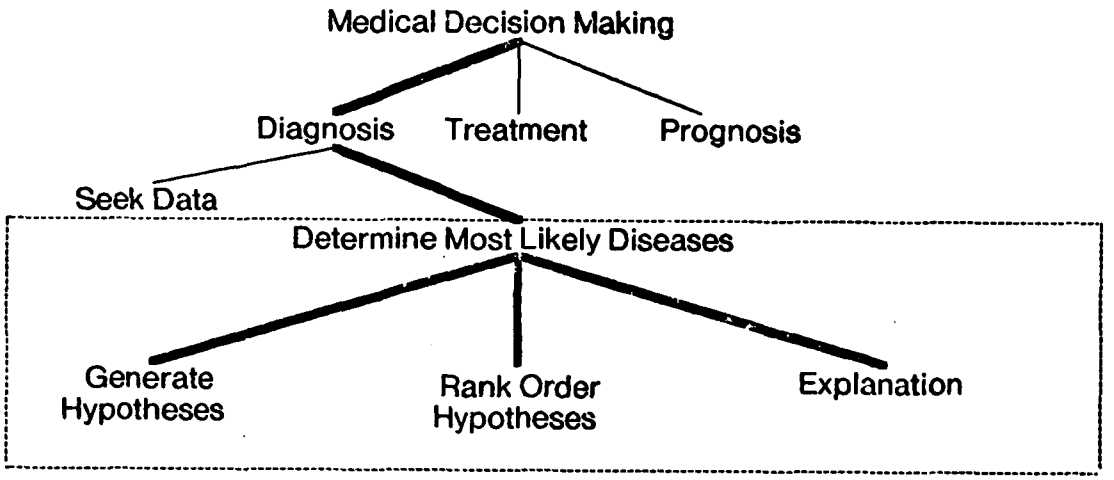


Figure 1-1: A Decomposition of the Tasks in Medical Decision Making

findings. This involves generating a set of candidate hypotheses and then rank ordering them. Although shown in the diagram as separate tasks, these two actually interact, so that generation and ordering occur incrementally. Explanation of the final ordering is also an important task. This enables NESTOR to explain why one hypothesis is considered more likely than another.

Note that in Figure 1-1 the the task of seeking data is not addressed by NESTOR. Thus, NESTOR is not concerned with strategies for data-seeking, but instead it focuses on how to determine the most likely diagnostic hypotheses given a particular set of findings(data). This is not because the question-asking task is considered unimportant, but rather that the two tasks are too large for a single dissertation to address in detail. In fact, seeking data and determining the likely diagnostic candidates usually work together in an iterative manner in any complete diagnostic system. The likely diagnostic candidates are

used by the data-seeking module in order to determine the next best finding to seek; once the finding is acquired the most likely diagnostic hypotheses are again determined; this continues until some condition to stop seeking data is satisfied. So, although NESTOR does not currently address data-seeking issues, it does perform the equally important task of locating the most likely hypotheses given a set of data; in the future this module can be interfaced to a data-seeking program module.

1.3. The Tasks Performed by NESTOR

NESTOR provides the physician-user with five basic commands:

1. **Accept a Set of Findings F**
   The findings can be multivalued.

2. **Accept a Diagnostic Hypothesis H**
   The hypothesis may contain multiple diseases.

3. **Generate the Best N Hypotheses**
   The program finds the best N hypotheses that account for the current set of findings, where the value of N is specified by the user. The user is given the ability to tailor the hypothesis generation process. For example, the user can specify particular diseases to be included or excluded from every hypothesis.

4. **Critique an Hypothesis H**
   H may be either a user supplied or a system generated hypothesis. The result is an explanation of how well H accounts for each of the current findings.

5. **Compare Hypotheses H_1 and H_2**
   This performs a comparative explanation of how well H_1 and H_2 account for the current set of findings. The pair of hypotheses can be a computer hypothesis vs. a computer hypothesis, a user hypothesis vs. a user hypothesis, or a user hypothesis vs. a computer hypothesis.

These commands allow the *user* to control the interaction rather than the program. A computer-based medical decision making aid in which the user has control will be called a medical decision support system (MDSS). In the next section the MDSS approach will be contrasted to previous interactive approaches.

1.4. NESTOR as a Medical Decision Support System

1.4.1. The Concept of a Medical Decision Support System

The ability of the user to control the decision making focus of the computer distinguishes an MDSS from most previous medical consultation systems (MCS) [Johnson 79]. An MCS typically constrains users to follow its problem solving focus by answering questions. In an MCS users may be allowed to request justifications for what the program has done, but seldom are they allowed to redirect the program's problem solving focus. In contrast, in an MDSS users have a high degree of control over the program. For example, hypotheses can be deleted from the differential diagnostic list being pursued, while others are added. The program can be instructed to analyze a specific diagnostic hypothesis at one point, then later instructed to suggest what it considers to be the most probable hypothesis. The important point is that an MDSS ideally gives the users as much control of the problem solving process as they desire. Because an MCS is an MDSS with minimal user control, an MDSS should be capable of performing as an MCS. Thus, the user may request that the MDSS operate in "system-initiated-dialogue-mode" for a period of time, in which case it would behave like an MCS.

One important reason for using the MDSS approach to CAMDM is that it is more responsive to the user's problem-solving focus than an MCS system; the program will do what the *user* wants, rather than control the interaction. In general, this added flexibility should improve user-acceptance. There is another important reason for using an MDSS. Individual physicians still know much more medicine and have more common sense than any computer program. This will remain true for some time. Additionally, physicians are human beings and can understand the human side of patient care. Computers, as we know them today, may never achieve this insight. So, the physician's insight is critical to the decision making process. Man has some strengths, the computer has others. An MDSS approach is designed to maximize the decision making symbiosis of the two.

1.4.2. The Historical Progression Toward Medical Decision Support Systems

The concept of an MDSS has progressively evolved over the last two decades. In the 1960's most CAMDM programs were limited to receiving patient data and returning a diagnostic label. The user had virtually no control of the interaction. In the 1970's this began to change. Research in developing decision support systems (DSSs) for business decision making began to increase [Keen 78]. A major advance toward MDSSs came when Shortliffe emphasized with MYCIN the importance of rule-based explanation in CAMDM [Shortliffe 74, Shortliffe 76]. Although users could not significantly direct the problem solving focus, they could at least carry on a dialogue with the program in order to understand its reasoning. A few years later Davis extended this line of explanation research [Davis 76]. His TEIRESIAS program aided users in augmenting or modifying its medical inference rules. This permitted alteration of the problem solving behavior of the program, but only *after* incorrect behavior was noted.

More recently, several programs have been developed that increase user control of the interaction. One deals with therapy selection for MYCIN [Clancey 78]. It can recommend a "best" therapy or successively "next best" therapies on demand. However, its most impressive MDSS feature is that the user can suggest a therapy and the program will critique it. Another program called ATTENDING [Miller 83] accepts as input a preoperative therapy plan, the patient's problems, and the planned surgical procedure. It then uses an augmented transition network representation in comparing the user plan to its own in order return a critique of the user plan. Along similar lines, a program developed by Langlotz [Langlotz 83] is able to critique a physician's oncology therapy plan for significant deviations from its own ruled-based recommendation. Unlike Clancey and Miller's programs, it can be directed to limit its critique to subparts of the plan. From a DSS standpoint, the greatest difference between NESTOR and the above three programs is that it deals with diagnosis and not therapy. Thus, an hypothesis about the present patient state is being critiqued rather than a plan for achieving some future patient state. Also, unlike the other three, NESTOR deals with causal and probabilistic knowledge in its assessment and critiquing procedures.

I am unaware of any *diagnostic* programs (as opposed to patient management programs) other than NESTOR that employ significant MDSS features. NESTOR can be viewed as one module of a total MDSS. Since diagnosis is a critical component of medical decision making and hypothesis evaluation is a critical component of diagnosis, it follows that important gains in MDSS research can be made by studying its application to computer-based hypothesis evaluation. The five user commands in the Section 1.3 summarized the basic MDSS capabilities of NESTOR. This is only a beginning; no claim is made that NESTOR contains all the MDSS features desired in a diagnostic program. Undoubtedly, future research will develop many improvements to both NESTOR's MDSS features and to the underlying diagnostic techniques that support those features. This research is viewed simply as one step in the study and development of computer-aided medical decision making.

In order to provide such a high level, interactive aid, specific technical problems had to be addressed. The next section will discuss the problems addressed and the techniques that were used to address them.

1.5. An Overview of NESTOR's Diagnostic Methods

NESTOR can be viewed as three primary program modules that are coordinated by a user interface module to provide the five user commands discussed Section 1.3. The three modules perform the following tasks:

1. *hypothesis scoring*

2. *hypothesis searching*

3. *explanation*

In this section each of the modules will first be discussed in terms of previous approaches and the problems with those approaches, then in terms of NESTOR's approach. Most of the systems discussed in this chapter are Bayesian programs [Miller 77, Warner 61, Wagner 78, Leaper 72, deDombal 74] and the artificial intelligence (AI) programs[2] CASNET [Weiss

[2]Winston provides a good introduction to artificial intelligence [Winston 84].

78], INTERNIST [Pople 75, Pople 77, Pople 82], MYCIN [Shortliffe 76, Buchanan 84], and PIP [Pauker 76].[3] These systems are well known and exemplify the most salient features of previous work in CAMDM. Other research that is more specific to each module will be reviewed in later chapters.

1.5.1. Scoring an Hypothesis

Figure 1-2 shows that the ability to score an hypothesis is central to being able to rank order a set of hypotheses.[4] Generally, the score is represented as a number such that the higher the number the more probable the hypothesis. Accurately scoring hypotheses is a difficult task and this topic has stimulated much previous research [Szolovits81a] [Shortliffe79] as well as the majority of the research effort in developing NESTOR.

1.5.1.1. Previous Approaches

Bayesian Programs

The application of Bayes' formula to medical decision making has been used extensively in the last 25 years. Figure 1-3 shows the formula, where F is a set of findings and H_i is an hypothesis from among N possible hypotheses. One of the primary advantages of this method is that it is formal. In general there are several benefits that result from using a scoring procedure based on formal probability theory:

1. Probability as a field of mathematics is well developed. This aids in designing scoring methods, in understanding how they relate to previous research, and in communicating them to other researchers.

2. Any assumptions made during the scoring of an hypothesis can be unambiguously expressed. This may aid the user in interpreting the resulting score. It may also help the designer of the scoring algorithm to determine which assumptions to make in designing the program.

---

[3] Szolovits and Pauker provide a good general discussion of these four AI programs and Bayesian programs [Szolovits 78].

[4] Cohen has recently experimented with categorical methods for rank ordering hypotheses [Cohen 83], but the generality of this approach is not known. Given the highly probabilistic nature of knowledge in medicine it seems unlikely that purely categorical methods will be adequate for general medical diagnosis, and therefore some means of scoring hypotheses will still be needed.

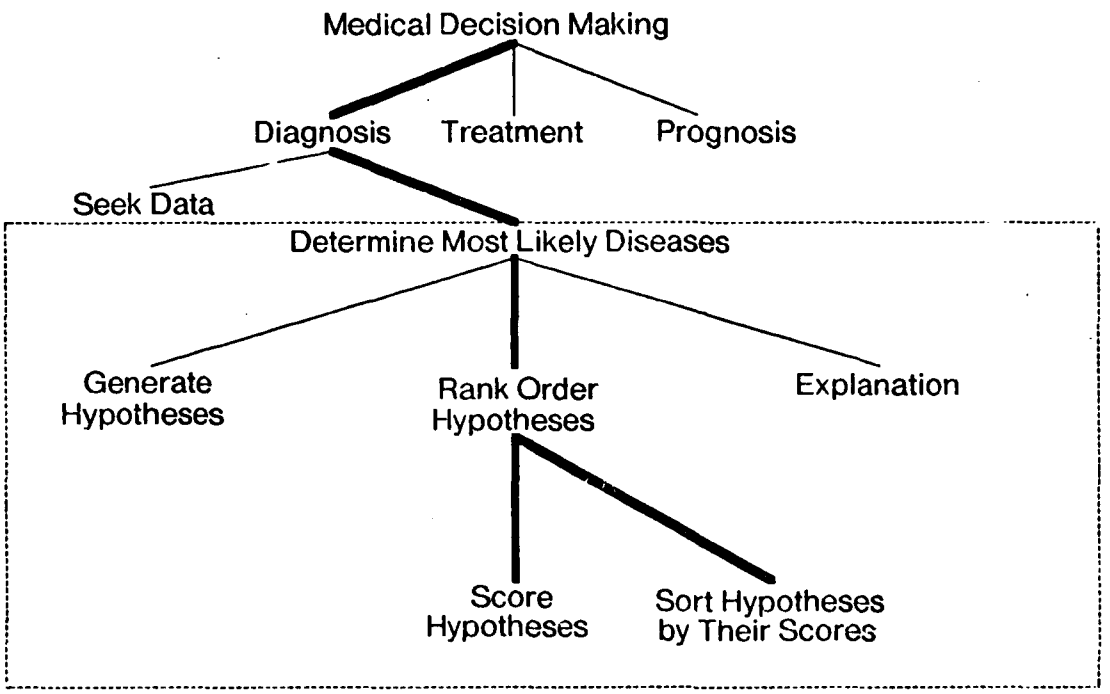


Figure 1-2: The Subtask of Scoring an Hypothesis

3. It is possible to utilize available statistical data *directly*.

4. It is possible to interface the hypothesis score (a probability) directly with other decision making procedures such as decision analysis programs [Schwartz 73, Weinstein 80].

Thus, there is a great attraction to using a formal system.

Implementations of Bayes' formula yield formal systems that provide a means of using the more readily available sensitivity probabilities (i.e., $P(F \mid H_i)$) rather than posterior probabilities (i.e., $P(H_i \mid F)$). However, they almost invariably make the following two assumptions [Szolovits78]:

$$P(H_1 \mid F) = \frac{P(F \mid H_1) \times P(H_1)}{P(F \mid H_1) \times P(H_1) + P(F \mid \text{not } H_1) \times P(\text{not } H_1)}$$

$$= \frac{P(F \mid H_1) \times P(H_1)}{\sum_{j=1}^{N} P(F \mid H_j) \times P(H_j)}$$

Figure 1-3: Bayes' Formula

1. *The conditional probabilities of a finding given an diagnostic hypothesis are independent.* This means that P(F | H) which is used in both the numerator and denominator of the last term in Figure 1-3 is approximated by $P(f_1 \mid H) \times \ldots \times P(f_m \mid H)$, where $f_1, \ldots, f_m$ are the findings in the set F.

2. *A diagnosis consists of a single disease from among a given set of diseases, which are exhaustive and mutually exclusive.* This is the assumption that the patient's clinical condition corresponds to one and only one disease from among a given set of diseases.

The problem with these assumptions is that often they are not valid. The first one is particularly questionable. Norusis implemented a system to diagnose cardiovascular diseases and found that assuming conditional independence caused a substantial increase in the misclassification rate [Norusis 75]. On the other hand, deDombal has developed a program that assumes conditional independence in diagnosing abdominal pain; in one study it was 91.5% accurate and actually performed better than senior staff physicians [deDombal 74]. One possible explanation for this discrepancy is that different domains differ in the degree to which findings are independent. At any rate, what seems clear is that independence can not be relied upon to yield accurate diagnoses across all fields of medicine under all possible conditions.

The second assumption is also often not valid, particularly in complex cases in which

multiple disease diagnoses are likely. It is just these complex situations in which a physician is most likely to seek a consultation.

Another weakness of most Bayesian programs is that they do not represent the imprecision of their scores. Instead, a single precise number is generated with no indication of its confidence limits. Thus, it is difficult to determine how to use the number.

In summary, most Bayesian programs have the advantage of a foundation in formal probability theory, but suffer from at least two flaws. First, the assumptions they make are often invalid. Second, they lack a measure of the imprecision of their scores.

AI Programs

A major difference between AI and Bayesian programs is that AI programs use a greater diversity of knowledge types that are more highly structured than a set of probabilities. Also, the types of knowledge are more often symbolic rather than numeric. The knowledge types may be either static, meaning the knowledge structures exist before any case is diagnosed, or dynamically created in the course of diagnosing a case. MYCIN for example structures its knowledge base by dynamically constructing a tree of rule invocations within a particular problem solving context. CASNET uses a causal graph to structure much of its knowledge-base. INTERNIST statically structures its diseases using an abstraction hierarchy, and also uses an algorithm to dynamically structure its differential diagnosis. PIP contains many types of links between diseases such as *caused-by*, *associated-with*, and *is-on-the-differential-diagnosis-of*. It also contains triggers which are used to invoke hypotheses. These links are used to structure the static knowledge base and to construct the differential diagnosis. A great deal of the power of these AI programs appears to lie in their more sophisticated knowledge representations (knowledge structures); their explicit representation of domain semantics constitutes one of their major methodological advantages over purely statistical systems.

A limitation of each of the four AI programs above is that they assume some degree of independence of the findings. However, unlike Bayesian programs, their independence

assumptions are usually more localized, as for example when CASNET assumes independence of effect nodes given their *direct* cause nodes. In general, local independence assumptions lead to less inaccuracy than global ones. Nevertheless, they are assumptions which may cause scoring errors.

A limitation shared by the AI and Bayesian approaches is that they represent the final score of a diagnostic hypothesis with a single number, thus giving no indication of the imprecision of the score.

Finally, perhaps the major disadvantage of these AI programs is that they all use an ad hoc scoring scheme. The strengths listed previously in describing a formal scoring system (see page 10) are precisely the weaknesses of the well-known AI programs. Of particular concern is the extensibility of the domain specific methods developed in each of the programs. All four programs have been shown to function well for the domain in which they were constructed and refined, but little is known about their extensibility to a broad range of medical domains. Of the four, INTERNIST yields the most useful insight into this question, because it already contains a large body of medical knowledge and the type of errors in its performance (and their causes) have been formally evaluated.

INTERNIST diagnoses diseases in internal medicine. Miller conducted a formal evaluation of the program in 1981 [Miller 82] and at that time it contained over 500 diseases, 3500 findings, and a total of about 70,000 conditional probabilities. The evaluation was based on 19 cases from the clinicopathological cases published in the New England Journal of Medicine. The accuracy of diagnosis was 58% for INTERNIST, 65% for the hospital clinicians, and 81% for the case discussants.[5] A detailed analysis was then performed to determine the origin of the program's errors, and it showed that 35% of them were due to correctable errors in the knowledge-base whereas 65% were due to faults in the design of the diagnostic program. These latter errors could be explained by the following five types of fault:

---

[5]For the purposes of the accuracy rating being used a correct diagnosis is considered one in which the correct disease is either stated or appears at the top of an unresolved differential list.

1. ***A Lack of Causal Knowledge***
   INTERNIST was not able to distinguish findings which were predisposing factors to a disease from findings caused by the disease. It was also not able to account for the pathophysiological dependencies between findings, and so it treated them as independent.

2. ***A Lack of Multiple Disease Hypotheses***
   Although INTERNIST can diagnosis multiple disease problems, it does this in a stepwise fashion by successively including in the hypothesis set the next "best" remaining disease on the differential until all findings are accounted for. It does not construct a multiple disease hypothesis and then score it.

3. ***A Weak Representation of the Severity of Findings***
   This is due to being unable to represent multivalued variables. All INTERNIST variables are binary.

4. ***No Representation of Temporal Knowledge***
   There is no general way in which the time at which findings occurred can be easily represented.

5. ***No Representation of Anatomical Knowledge***
   There is no general way in which the location of a patient's symptoms and signs can be easily represented, or that a diagnosis can include the location of a particular disease process.

Summary

Combining the strengths of the Bayesian and AI approaches while avoiding the specific weaknesses discovered in the INTERNIST program suggests that a scoring methodology should have at least the following features:

1. A formal probabilistic foundation

2. A representation of the imprecision of a score

3. A rich structuring of knowledge, in particular the use of causal knowledge when available

4. The ability to score multiple disease hypotheses

5. The ability to represent and reason with severity knowledge

6. The ability to represent and reason with temporal knowledge

7. The ability to represent and reason with anatomical knowledge

NESTOR addresses the first five of these seven concerns. The techniques it uses to do this will be discussed in the next section.

1.5.1.2. NESTOR's Approach

Using a Formal Probabilistic System

NESTOR is a formal probabilistic system. This means that the score assigned to an hypothesis is a formal probability. In most previous probabilistic systems the assigned score is the posterior probability of the hypothesis H given the set of findings F, that is P(H | F). However, NESTOR scores hypotheses by determining P(F & H). This simply expresses the probability of the findings and the hypothesis co-occurring. Chapter 4 contains a proof showing that this ***scoring metric*** is computationally much easier to derive than P(H | F), and that it is sufficient to rank order a set of hypotheses.

Since P(F & H) = P(F | H) x P(H), the computation of P(F & H) may be subdivided into determining P(F | H) and determining P(H). Figure 1-4 shows how the hypothesis scoring subgoal can be divided into these two tasks. These are the same type of probabilities that are used in the application of Bayes' formula. In fact, the utility of Bayes' formula is due to the *availability* of these probabilities (sensitivity and prevalence information) as opposed to posterior probabilities. Conditional probabilities in medicine are largely available as sensitivities rather than posterior probabilities. Thus, the literature is more likely to have data in the form of P(F | H) than the form P(H | F). Additionally, physicians are more comfortable relating subjective estimates of P(F | H) than of P(H | F). In a like manner, P(H) is commonly either available in the literature or can be estimated by a physician.

The computation of P(F | H) is unquestionably the most difficult computation that NESTOR must make. However, notice that the direction of this conditional probability

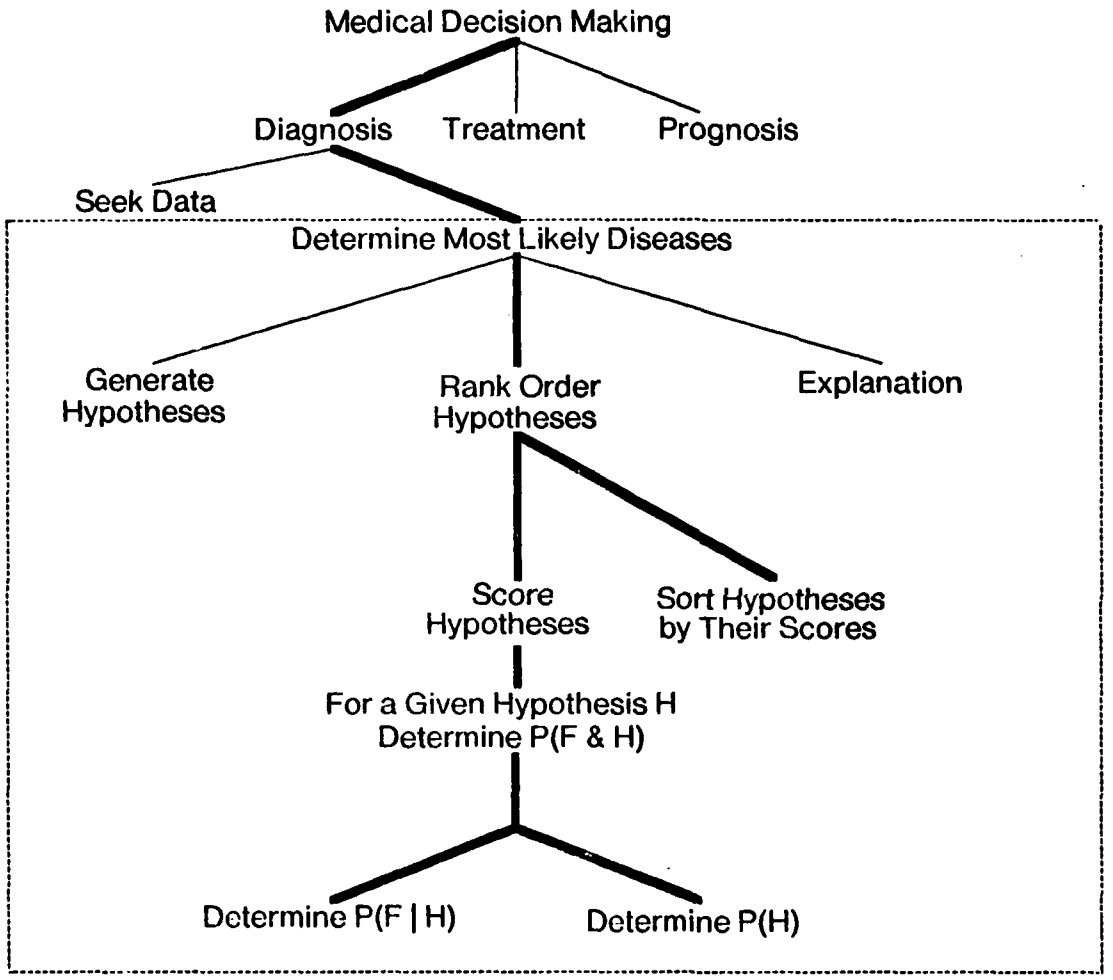


Figure 1-4: The Subtasks of Computing P(F | H) and P(H) in Scoring an Hypothesis

from hypotheses (i.e., the etiologies of a set of diseases) to findings makes the application of causal knowledge quite natural since it too is directed from the etiologies to the findings. Recall that the use of causal knowledge was listed in the last section as one desirable feature of a diagnostic system. It is helpful that we now have a probability which is readily amenable to the use of causal knowledge. Shortly, we will see precisely how such knowledge is used in computing P(F | H).

Scoring Multiple Disease Hypotheses

An hypothesis in NESTOR can contain an arbitrary number of diseases. The hypothesis is scored as a whole, rather than one disease at a time. Thus, NESTOR avoids the problems encountered by stepwise construction of hypotheses.

Bounding Probabilities

Probabilities in NESTOR are viewed in terms of the frequency theory of probability [Swinburne 73]. In this theory probabilities are considered as statements about the relative frequency of events in the past. NESTOR assumes that the past frequency of an event is predictive of the future frequency of that event. In this way, knowledge of the past occurrence and presentation of diseases can be used to predict their present likelihood of occurrence and presentation, and as discussed earlier in this section, this can be used to determine the most likely diseases given the findings.

There are two primary sources of the probabilities in NESTOR's knowledge-base: expert estimates and database statistics. An expert estimate is assumed to be a prediction by an expert of the frequency of an event in the the patient population being served.

NESTOR represents the probability of an event E with an upper bound (UB) and a lower bound (LB), and terms this range $P_B(E)$, so $P_B(E) = (P_{LB}(E), P_{UB}(E))$. The *precision* of $P_B(E)$ is defined as $1 - (P_{UB}(E) - P_{LB}(E))$. Thus, a point probability has a precision of one, whereas the probability range of (0, 1) has a precision of zero. $P_B(E)$ is said to be *valid* if it includes P(E), the frequency of event E in the population.

Most previous diagnostic programs have represented uncertainty as a single number, thus fixing its precision. In the case of some Bayesian programs the precision is

unrealistically high in light of the population data available. For AI programs such as INTERNIST the precision is fixed but low, since there are only five numbers used to specify conditional probabilities. The problem with fixed precision, whether high or low, is its inherent inflexibility. Ideally, a diagnostic system should allow flexible expression of the precision of the probabilities in its knowledge base, dependent on the degree of confidence in the value of the probability.

By representing probabilities with bounds, NESTOR allows the experts who are building the knowledge base to adjust the precision of their subjective probability estimates on the basis of their confidence in them. Experts are more likely to be comfortable expressing the probability of the finding f given disease d as being between 40% and 60%, than to have to specify it as 50% exactly. A probability range is also more likely to be valid than a point probability.

In addition, the ability to represent probabilities is shown in Section 4.3.3[6] to be important in computing valid joint conditional probabilities which are critical to the process of scoring hypotheses. In computing the score of an hypothesis, NESTOR attempts to maintain as much precision as possible given the currently available knowledge-base, while still retaining validity of the score.

The use of *bounded* hypothesis scores means that sometimes hypotheses can only be *partially* ordered, due to an overlapping of their scores.

The Causal Structuring of Probabilities

NESTOR has been designed to work in domains in which the pathophysiological basis of the diseases are well understood. Thus, the causal relationships between the etiologies and the findings are assumed to be known. This does not mean that the detailed mechanism between nodes in the graph are necessarily known or represented, but only that all nodes in the graph which have a significant causal influence on a node also have a pathway leading to that node.

---

[6]This notation will be used throughout the dissertation. The 4 in 4.3.3 indicates that the section appears in Chapter 4, that is, within Chapter 4 there is a section that is labeled 4.3.3

Earlier in this section it was mentioned that causal knowledge is often useful in calculating P(F | H). NESTOR uses causal knowledge to ***structure*** individual conditional probabilities so that P(F | H) can be more accurately calculated. There is a great deal of information in the causal structure itself. For example, Figure 1-5 shows a case in which the joint conditional probability (JCP) of two findings are computed given a single disease etiology.

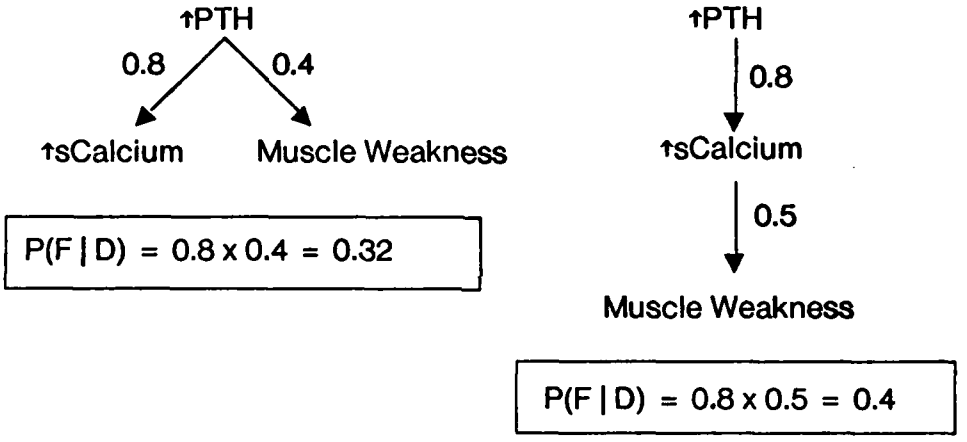


Figure 1-5: Using a Conditional Independence Assumption vs. Causal Knowledge in Calculating a Joint Conditional Probability

In this case the disease (D) is primary hyperparathyroidism (PHPT), the etiology is an increase in the hormone PTH (due to uncontrolled overproduction of PTH which is uniformly observed in PHPT), and the findings (F) are an increased serum calcium (f_1) and muscle weakness (f_2). The arrows in the figure indicate causality and the numbers beside them indicate the probability that the *effect* will be present when the *cause* is present. On the left side of the figure the JCP is calculated assuming that the two findings (f_1 and f_2) are

conditionally independent.[7] This is a common assumption made in most Bayesian and AI diagnostic programs. However, the right side of the figure shows that f_1 and f_2 are in fact causally related. This allows the accurate calculation of the JCP as 0.4 instead of 0.32. Notice that the problem with the calculation on the left is that by assuming independence it "counts" the occurrence of increased serum calcium twice: once explicitly (0.8) and once implicitly in the probability of muscle weakness (*0.8* x 0.5). By correctly representing the causal interactions, this problem is avoided. The difference between 0.4 and 0.32 may seem slight, however, the ratio of the correct score to the erroneous one increases exponentially as a function of the depth of the causal chain of findings. Thus, the magnitude of the error can be significant even in causal graphs of only a few levels depth.

Causally structuring conditional probabilities also generally decreases the amount of knowledge that must be acquired to represent the complete joint conditional probability space of a disease. The left of Figure 1-6 shows a flat representation of the JCP space of 8 findings given the disease etiology. If we assume for simplicity that the variables are binary, then there are a total of $2^8 = 256$ different combinations of the findings for the given etiology. Thus, the size of the complete space of conditional probabilities is 256. In contrast, the right side of the figure shows a possible causal structuring of the hypothetical disease with the etiology at the top, 6 intermediate nodes in the middle, and 8 findings at the bottom. In general some of the intermediate nodes may be finding variables with currently unknown values, while others may be states or processes that are almost never directly observable. Since the nodes are binary, the JCP space of each branch point can be represented by $2^3 = 8$ probabilities. Furthermore, since there are 7 branch points, a total of $8 \times 7 = 56$ probabilities must be represented. Even in this small example, a substantial reduction of 200 probabilities has been achieved. This represents 200 probabilities that an expert who is building the knowledge base would not have to specify. In general if there are n findings and B is the maximum branching factor in the causal graph, then the reduction is from $O(2^n)$ to $O(n \times 2^B)$. Since B is usually much smaller than n, the causal structuring of

---

[7] Conditional independence between two findings f_1 and f_2 occurs when in the context of some disease D the presence of finding f_1 does not influence the probability of f_2 given D, and similarly the presence of f_2 does not influence the probability of f_1 given D. In mathematical terms, f_1 and f_2 are conditionally independent given some disease D if and only if $P(f_1 \& f_2 \mid D) = P(f_1 \mid D) \times P(f_2 \mid D)$. For N findings, f_1 to f_N, the equation is generalized to $P(f_1 \& f_2 \& \ldots \& f_N \mid D) = P(f_1 \mid D) \times P(f_2 \mid D) \times \ldots \times P(f_N \mid D)$.

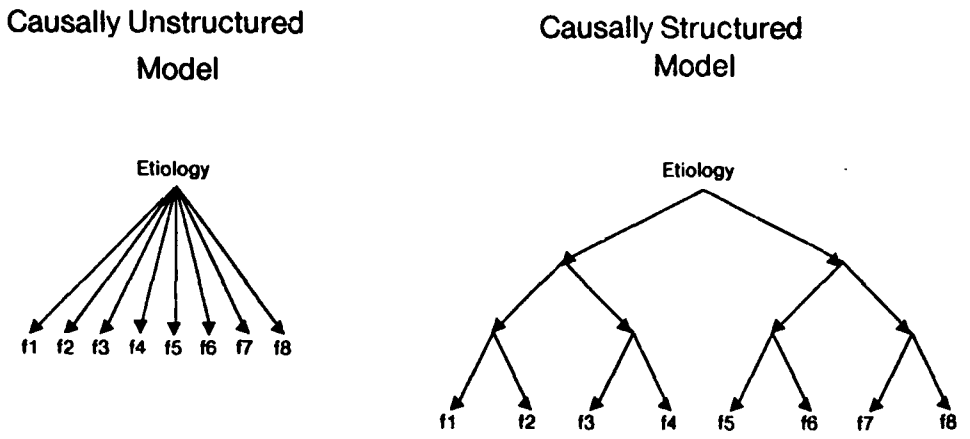


Figure 1-6: An Example of Causally Unstructured and Causally Structured Models

conditional probabilities will typically lead to a significant reduction in the number of probabilities required to specify and store the complete JCP space.

Probabilistic Causal Simulation

Causal structuring of probabilities has been shown above to be potentially useful, yet some technique is needed to use the causal structure to calculate P(F | H). NESTOR uses a method which will be called probabilistic causal simulation (PCS). The goal is to determine the probability that the etiologies E of the diseases in the hypothesis H cause the findings F. This is expressed as P(F | E) and is equal to P(F | H). The PCS method may be understood intuitively as follows: it determines *every* possible unique way in which the given etiologies E can cause the findings F[8] ; the sum of the likelihood of all such

[8]This is of course subject to the limitations of its causal knowledge about the mechanisms linking E and F.

possibilities equals the probability that the given E is causing F. The process of determining "every possible unique way in which a given E can cause F" involves simulating the possible causal pathways between the etiologies and the findings.

Figure 1-7 shows a causal graph that will be used to illustrate a PCS.

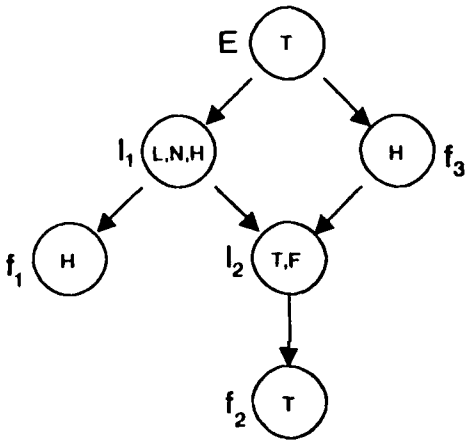


Figure 1-7: The Graph Used in a Probabilistic Causal Simulation

The names of the variables are outside the circles which enclose their values. In this example the variables are either binary with a value T or F, or tertiary with values H(igh), N(ormal), or L(ow).[9] E is the etiology node, and f_1, f_2, and f_3 are the findings. I_1 and I_2 are two intermediate causal nodes, and each is either a finding with a currently unknown value or a variable for which values are almost never directly available.

[9]Since the values of any node in the causal simulation may be multivalued, this addresses one of the problems found in INTERNIST, namely the lack of a natural representation for the severity of findings or etiologies.

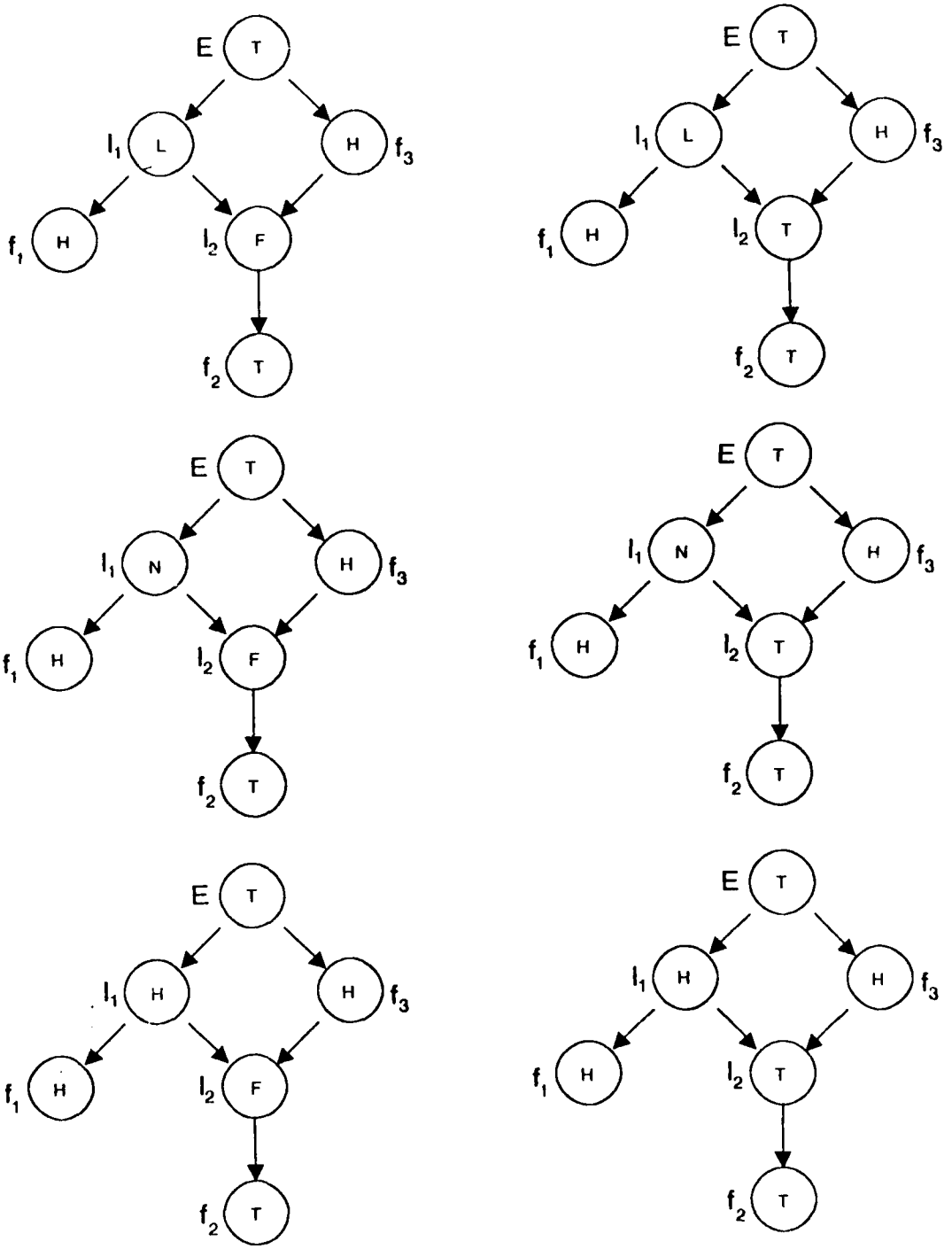


Figure 1-8: The Six Instantiations of the Graph in Figure 1-7

Notice that the values of the etiology and the findings have known values, while the values of I_1 and I_2 do not. Instantiating every possible combination of values for I_1 and I_2 yields the six graphs shown in Figure 1-8. These six graphs represent every way in which the etiology can possibly affect the findings. Each of these graphs is called an ***instantiated graph***. The probability of an instantiated graph represents the likelihood that the given etiology will cause the findings via the particular set of values of the intermediate nodes in that graph. The sum of the probabilities of all six graphs equals the probability that the given E is causing f_1, f_2, and f_3.

The computation of the probability of an instantiated graph is the most difficult problem that NESTOR must address in scoring an hypothesis. The key computation is to determine the probability of a set of effects given their immediate causes. This is called a local joint conditional probability (local JCP) calculation and will be discussed in the next section. The details of the probability assessment of an instantiated graph are discussed in Chapter 4. The important point for now is to realize that it is not sufficient to represent the problem as a *single* instantiated graph in which probabilistic inferenceing begins with the etiology at the top and proceeds to the findings at the bottom. Instead some method is needed for combinatorially exploring each possible instantiation of those nodes with uninstantiated values. Only a graph that contains fully instantiated node values, such as those in Figure 1-8, can be scored.

One major advantage of the simulation methodology is that it makes no general assumptions about the conditional independence of the nodes in the graph. Instead, any such independence assumptions are made when computing the local JCP's in an instantiation graph. Thus, independence assumptions are not imbedded in the inference procedure itself. This separation of the inference assumptions from the inference procedure is similar in purpose to the common AI notion of separating the inference procedure from the knowledge base. Both seek to separate out those aspects of a problem solving system which are conceptually and functionally independent of each other.

In contrast, consider the AI systems CASNET [Weiss 78] and PROSPECTOR [Duda 76]. Both use causal knowledge to structure the application of conditional probabilities. However, the inference procedure of both systems makes a general assumption about the

conditional independence of the probabilities in the causal graph. This assumption may be reasonable for some domains, but for others it may not. The flexibility to locally represent portions of the graph in which *dependence* is known to exist has been lost. Additionally, if the causal graph is a lattice instead of a tree, then the independence assumption is necessarily violated, regardless of the domain. Duda acknowledges this assumption in PROSPECTOR in saying that "if the network contains multiple paths linking a given piece of evidence to the same hypothesis, the independence assumption is obviously violated" (see [Duda 76], page 1080).

The causal simulation technique used by NESTOR in scoring hypotheses is more general than the other inference techniques just discussed, however, it is also more computationally expensive. Although the combinatorics of causal simulation may seem to become quickly intractable in a nontrivial graph, Section 4.6.1 will discuss some caching methods in which the computation time can often be made to increase only linearly as a function of the number of nodes in the graph.

Calculating Local Joint Conditional Probabilities

Once an instantiated graph is created, the probability of the findings given the etiology must be calculated for that graph. This can be accomplished if the local JCP's between adjacent nodes in the graph can be calculated. Figure 1-9 shows one instantiated graph at the top (taken from the graph in the top right corner of Figure 1-8), and below it are the three JCP's necessary to calculate its probability. If P_1, P_2, and P_3 can be calculated, then the probability P(F | H) can be calculated as $P_1 \times P_2 \times P_3$. The calculation of such JCP's is similar to the situation faced by a simple Bayesian program. In those programs, as in NESTOR, it is common to know the individual conditional probabilities between one cause node and its effect node, but less common to know a joint conditional probability between a cause and several of its effects. If there is not any additional knowledge to use in making the calculation, then NESTOR's calculation of the JCP is identical to a Bayesian calculation.[10]

Even in such a case, this should not be interpreted to mean that the overall calculation of

---

[10] This is true if the assumption set NESTOR is using specifies that no knowledge of dependence should by default be taken as the existence of independence. Section 4.3.3.1 discusses NESTOR's use of different sets of assumptions in scoring hypotheses.

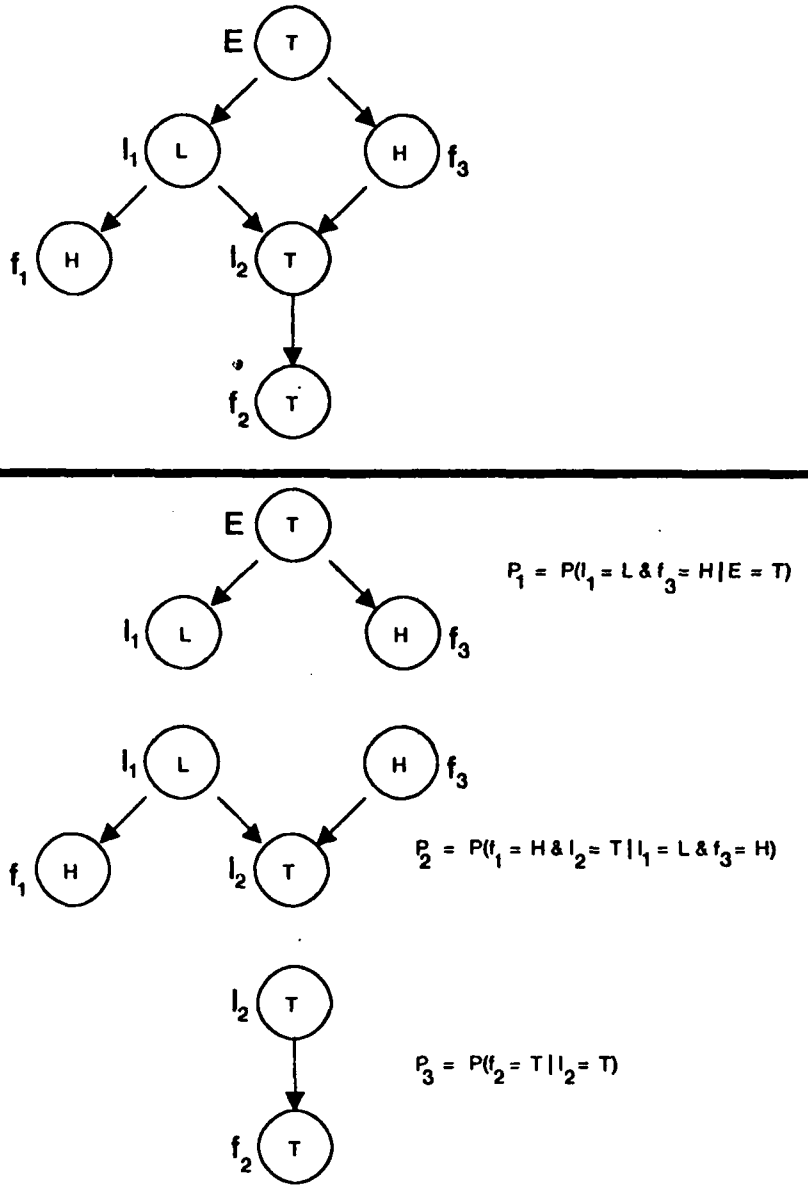


Figure 1-9: Joint Conditional Probabilities in a Causal Graph

the probability of the graph is no better than a Bayesian approach that uniformly assumes independence and uses no causal knowledge. The causal knowledge has structured the probabilities so that overall, fewer independence assumptions are being made.

If there *is* additional knowledge regarding the relationship of the variables in the local JCP, this can be used to improve the accuracy of the local JCP calculation. NESTOR has two types of causal information: probabilistic and categorical[Szolovits78]. Only the probabilistic type has been discussed so far. The categorical knowledge contains information about the functional form of the relationships between cause and effect nodes. Currently in NESTOR the expression of functional form is limited to stating the monotonicity of the relationship between a cause node and its effect. Often by combining individual ***conditional probabilities*** with knowledge of ***functional form*** a local JCP can be ***bounded*** using reasonable ***assumptions***.

Figure 1-10 shows an example. The solid arrows represent categorical causal relationships (also called functional relationships) for which no conditional probabilities are known. The broken arrows are the known conditional probabilities between nodes, which in the example are 0.5 and 0.4 . The + symbol indicates that the effect variable is a monotonically increasing function of the cause variable, although the exact functional form is not necessarily known. The - symbol indicates a monotonically decreasing relationship. I_1, I_2, and I_3 are intermediate causal nodes. The goal is to calculate the following JCP: P(increased E_1 & increased E_2 | increased C). A simple approach NESTOR could take would be to assume conditional independence and calculate the JCP as 0.5 x 0.4 = 0.2. The problem with this is that the independence assumption may be invalid. At the other extreme, NESTOR could make no assumptions at all and bound it as $0.4 \geq JCP \geq 0$. The upper bound is 0.4 since the likelihood of a joint event can be no greater than the likelihood of its least likely member, which in this case is 0.4 . However, if possible, it is preferable derive a tighter lower bound than zero. In this case there is additional knowledge available in the form of functional relationships. If NESTOR can show that given *increased C* then *increased* E_1 and *increased* E_2 are not negatively correlated, then assuming *increased* E_1 and *increased* E_2 to be conditionally independent given *increased C* will form a lower bound. Inspection of the monotonic relationships shows that E_1 and E_2 share no causal pathway

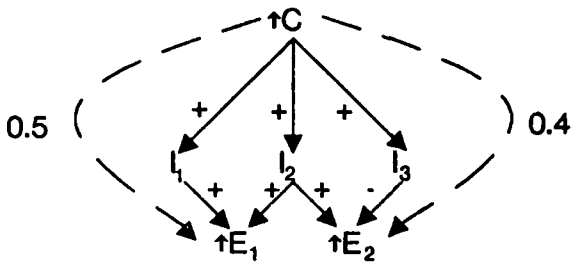


Figure 1-10: Using Categorical and Probabilistic Causal Knowledge to Calculate a Joint Conditional Probability

independent of C which includes a decreasing function. Therefore, given *increased C*, the occurrence of *increased* E_1 does not decrease the likelihood of *increased* E_2, nor does *increased* E_2 decrease the likelihood of *increased* E_1. Thus, assuming them to be conditionally independent serves as a lower bound on this JCP. Notice that assuming independence in calculating this lower bound is different from making a direct assumption of independence of the JCP. First, only the lower bounds of the JCP, not its exact value, is calculated by assuming independence. Second, available functional knowledge is being used to support the assumption of independence for the lower bound.[11] So, the final result is $0.4 \geq JCP \geq 0.2$.

The important point to remember is that knowledge of the functional form of causal relationships can sometimes be combined with sparse probabilistic information to derive accurate bounds on a JCP while using reasonable assumptions.

[11] Other calculations discussed in Chapter 4 are used when the causal relationships indicate that the findings may be negatively correlated

Even in domains in which all JCP's are known for every *individual* disease, there is still the need to calculate bounds on JCP's which are created when two or more disease graphs are combined during the process of scoring multiple disease hypotheses. So, the above techniques will still be relevant.

1.5.2. Searching for the Most Probable Hypothesis

A major task of any computer-based medical diagnostic aid is to determine the most likely diagnostic hypothesis to explain a given set of findings. Determining the most likely hypothesis among many possible hypotheses requires some means of generating hypotheses and then testing the likelihood of those hypotheses in accounting for the findings. The task of testing the likelihood of hypotheses corresponds to the scoring task discussed in Section 1.5.1. The task of integrating the generation and scoring of hypotheses in order to find the most likely hypothesis is the subject of this section. This task will be called hypothesis search.

1.5.2.1. Previous Approaches

Most implementations of Bayes' formula have simplified the task of hypothesis search by assuming that there is only one ongoing disease process in the patient.[12] The most likely hypothesis is determined by computing the posterior probability of every disease and selecting the one with the highest score. The advantage of this approach is that the number of hypotheses that must be scored is relatively small compared to the situation in which multiple-disease hypotheses are considered as possible. The disadvantage is that single-disease hypotheses are inadequate when the patient has more than one disease.

The AI CAMDM programs CASNET, INTERNIST, MYCIN, and PIP *are* able to diagnose more than one disease for a given patient. However, each of them constructs a multidisease hypothesis in a stepwise fashion; at any given step only single-disease hypotheses are scored. The advantage of this approach is that multidisease hypotheses can be generated to explain complex cases. The disadvantage is that the methods are ad hoc and

---

[12] Section 5.6 of Chapter 5 discusses methods of diagnostic hypothesis search in more detail.

subject to potential error; in particular, multiple-disease hypotheses are never scored as a unified pathophysiological process. This can be especially undesirable when the pathophysiology processes of the diseases in an hypothesis causally interact.

1.5.2.2. NESTOR's Approach

NESTOR scores multiple-disease hypotheses as a unified pathophysiological process. The advantage of this approach is that the causal interactions of the diseases in an hypothesis are explicitly considered. The disadvantage is that the number of multiple-disease hypotheses is very large. For a knowledge-base containing 100 diseases, there are more than 10^{30} multiple-disease hypotheses. Clearly some method is needed for limiting the number of hypotheses that must be generated and scored. In NESTOR this is done by using a search method that limits the hypotheses that are generated on the basis of their prior probabilities.

NESTOR uses a branch and bound search technique to limit hypothesis generation. This technique is able to disregard provably non-optimal areas of the search space. To understand the general notion of the technique, consider the task of finding the shortest route from a home city H to a destination city D. If a route from H to D of 10 miles is already known, then any route beginning at H and going through an intermediate city I that is more than 10 miles away from H can be ignored, since it is already worse than a known route of 10 miles. The savings results from not having to search for a route from I to D. This is a method of pruning (i.e., eliminating branches from) the search process.[13] In NESTOR a similar technique is used with diseases for cities and reciprocals of probabilities for distance, with the goal being to find a *pathway* of diseases (not necessarily a *single* disease) which explains all the findings with maximum probability (minimum distance). The pruning occurs when the extension of *an hypothesis* HYP (*routes from city H*) to include a particular *disease DX* (*city I*) can be proven to be always *less probable than* (*a longer distance than*) a currently known *hypothesis* (*route*) given the current findings. In this case the extension of the *hypothesis* HYP (*route from H*) containing *DX* (*city I*) can be pruned

---

[13] Note that the shorter the currently known *best* route, the more pruning that is possible. A known route from H to D of only 5 miles would generally allow more pruning than a route of 10 miles.

(ignored) because it is necessarily suboptimal. Chapter 5 discusses the details of this search technique. Section 5.3 contains a proof that the technique is admissible, that is, it is guaranteed to find the most likely hypothesis.

NESTOR also allows the user to have significant control over the hypothesis search process.[14]

- The user can specify a number N, where N is the number of top hypotheses to be determined.

- A time limit can be set on how much real-time NESTOR will expend in searching for the N most likely hypotheses. If the time-limit is reached, then NESTOR will display the N most likely hypotheses it has found so far.

- Specific diseases can be excluded or included in every diagnostic hypothesis that NESTOR generates. This allows the user to guide the hypothesis search process on the basis on knowledge or data that might not be available to NESTOR.

1.5.3. Explanation

1.5.3.1. Previous Approaches

A review of previous approaches to CAMDM explanation are discussed in two other sections. Section 1.4.2 (on page 7) discusses previous approaches to CAMDM explanation from the standpoint of their medical decision support characteristics. Section 6.3 (on page 212) discusses previous approaches that have used explanation techniques that are technically similar to NESTOR's methods.

1.5.3.2. NESTOR's Approach

NESTOR is designed to explain the likelihood that a *given* hypothesis causally accounts for a *given* set of findings. Currently, NESTOR does not suggest *additional* findings to seek. Thus, it is not yet programmed to explain the reasons *why* it would be useful to determine the value of a particular finding. Instead, NESTOR focuses on explaining *how* an hypothesis can account for a given set of findings.

---

[14] See Section 5.5 for a complete discussion of the options and how they are implemented.

NESTOR can either COMPARE two hypotheses or CRITIQUE a single hypothesis. The COMPARE command instructs NESTOR to contrast how well two diagnostic hypotheses account for the current set of clinical findings. The key feature of NESTOR's explanation method is its use of causal knowledge. The causal knowledge that was used in scoring the two hypotheses is also used in generating the explanation of how they compare. The hypotheses being compared can be either a user hypothesis vs. a user hypothesis, a computer-generated hypothesis vs. a computer-generated hypothesis, or a user hypothesis vs. a computer-generated hypothesis. Thus, for example, users can formulate a number of their own hypotheses and have them compared to computer generated hypotheses which have been created under differing user-specified constraints. In this way NESTOR provides a flexible diagnostic tool that allows the user to experiment with many ideas about the etiology of a patient's problem.

The COMPARE task consists of two stages. First, NESTOR prints an English text version of the possible causal relationships linking the etiologies of the respective diseases in the two hypotheses to the findings in the current case. This case-specific causal knowledge gives the user a context in which to interpret the next stage of the comparison.

The second stage of the comparison informs the user how each of the current findings influences the relative probabilities of the two hypotheses. This complements the previous output of the qualitative causal structure of the hypotheses by giving the user a quantitative, probabilistic sense of their causal differences.

The CRITIQUE command is similar to the COMPARE command, except that a single hypothesis is abstractly critiqued with respect to *all* other diagnostic possibilities rather than just one other hypothesis. The hypothesis being critiqued can be either user-generated or computer-generated.

In critiquing an hypothesis H with respect to all potential competitors, it is useful to know the *posterior probability* of H given the findings. For example, an hypothesis may be provably more probable than all the others, but if it has a probability of only 2%, then there would be little confidence in it.

The calculation of the posterior probability of an hypothesis can be expressed as follows:

$$P(H_i \mid F) = \frac{P(F \,\&\, H_i)}{\sum_{j=1}^{N} P(F \,\&\, H_j)}$$

where there are N exhaustive and mutually exclusive hypotheses.

One of the original reasons for choosing P(F & H) as a scoring metric in NESTOR is that hypotheses can be ordered according to their likelihood without the expensive calculation of the denominator in the above equation (see Section 5.3 for a proof of this). The reason that the precise calculation of the denominator is so difficult is that it involves summing over *every* possible hypothesis. For 100 diseases, when all disease combinations are considered, this leads to over 10^{30} possible hypotheses. Clearly, this is not a computationally tractable approach. NESTOR avoids this problem by abandoning the goal of a precise calculation of the denominator and instead uses a method that can rapidly calculate its upper and lower bounds (see Section 6.2 for details). Using the bounds of the denominator and the bounded score of the hypothesis H_i being critiqued (i.e., $P(F \,\&\, H_i)$), the above equation can be used to calculate the posterior probability of the hypothesis H_i given F.

The explanation task is currently the least developed area in NESTOR. The primary goal has been to begin to investigate how causal and probabilistic knowledge, which are used for scoring hypotheses, can also be used to explain them. The current state of NESTOR's explanation methods have suggested several areas for future research and these are discussed in Section 6.4.

1.6. The Domain of Application

NESTOR has been implemented to diagnose the major causes of hypercalcemia. Hypercalcemia is a condition in which there is an abnormally high concentration of calcium in the blood [Myers 77, Lee 78, Petersdorf 83]. This has potentially serious consequences including death, and therefore determining its underlying cause is important. NESTOR

represents seven common causes of hypercalcemia, which are primary hyperparathyroidism, non-metastatic cancer, metastatic cancer, myeloma[15], Grave's disease, pheochromocytoma, and sarcoidosis. When these diseases are used in examples in later chapters, the necessary medical knowledge to understand them will be presented.

There are several reasons why hypercalcemia was chosen as a domain.

1. *It is a common, well defined clinical problem which is clinically important.* This aids in finding test cases, and in evaluating NESTOR's performance on them.

2. *The causal mechanisms of the diseases causing hypercalcemia are relatively well understood and diverse in type.* This is important since the design of NESTOR is intended to explore how causal knowledge of diseases can be useful in diagnosis.

3. *The domain is small enough to be implemented within the time frame of a dissertation.* It is important to realize that hypercalcemia is primarily being used as a small test domain for concepts and techniques that are intended to be applicable to much larger areas of medicine.

4. *A collaborating endocrinologist was willing to help in constructing the causal models of the diseases in this domain.*

1.7. Design Assumptions

The primary assumptions in the design of NESTOR are as follows:

1. *The diseases diagnosed are causally well understood.*
   This is really not a necessary assumption, but more a preferred condition if NESTOR is to use its techniques to improve the accuracy of diagnosis over programs that uniformly assume independence. In fact, if NESTOR has no causal knowledge, then it can be made to behave exactly like a Bayesian

[15] Although myeloma is a form of cancer, it is represented explicitly in NESTOR because of its interesting clinical presentation and pathophysiological mechanisms. Non-metastatic cancer and metastatic cancer are defined as excluding myeloma

program that uniformly assumes conditional independence[16]. So, the addition of valid causal knowledge can only improve the accuracy of NESTOR's scoring procedure, and in no case will it be less accurate (valid) than a Bayesian program that uniformly assumes conditional independence.

2. *Cause/Effect Relationships are Instantaneous.*
   In order to avoid the complex issues of temporal representation and reasoning, NESTOR assumes that the only causal relationships being modeled are those in which the value of an effect is influenced only by the current value of its cause. In Section 4.6.6 some proposed extensions to this simplification are discussed.

3. *Conditional and prior probabilities are available.*
   The causal simulation technique depends on the existence of conditional probabilities between causal nodes. The lack of these will decrease the precision of the diagnostic scores, but not their validity. The existence of prior probabilities (prevalence) of diseases is also important in maintaining the precision of diagnostic scores; in addition, they are necessary for the operation of the search technique that is used in locating the most likely diagnostic hypothesis. The prevalence of diseases is of course population-specific, as are some conditional probabilities. Thus, NESTOR assumes that its knowledge-base represents probabilities that are specific to the population being diagnosed.

4. *Findings are correct*
   NESTOR currently assumes that there are no erroneous findings. Section 4.6.2 discusses a method of eliminating this assumption by encoding additional knowledge using the current knowledge representation.

1.8. Overview of Chapters

In Chapter 2 a number of examples will be used to demonstrate NESTOR's capabilities and to show the operation of the user interface.

Chapter 3 presents the knowledge representation structures in NESTOR, which include diseases, findings, intermediate causal nodes, categorical causal links, and

---

[16] Whether it does this or not depends on the set of assumptions NESTOR is given to use in such a case. An alternative to assuming conditional independence, would be to make no assumptions and to bound the probabilities using formal mathematical relationships. However, in general this would lead to very wide bounds.

probabilistic causal links. The procedure that allows a user to enter findings will also be discussed.

Chapter 4 contains a detailed description of the hypothesis scoring procedure. The techniques for using causal knowledge in diagnostic scoring are discussed extensively.

Chapter 5 presents the hypothesis search procedure. This module uses the scoring procedure plus a branch and bound search technique to locate the most probable diagnostic hypothesis.

In Chapter 6 the explanation procedure is discussed. This procedure can critique a single hypothesis, or compare two hypotheses.

Finally, Chapter 7 contains a summary and the conclusions.

Chapter 2

Examples

In this chapter several examples will be used to demonstrate how a user interacts with NESTOR. The emphasis will be on showing a varied range of commands and responses, rather than focusing in-depth on one clinical case. The examples are kept simple in order to quickly and clearly convey the principles of NESTOR's operation. The reader should keep in mind that the current knowledge-base is limited to seven diseases, and therefore possible diagnoses are limited to some combination of this set.

In what follows the user's input is underlined; italics are used for comments on the interaction; the plain text was produced by NESTOR.

(NESTOR)

The user is in the LISP environment and requests to use NESTOR. The first task is to enter a set of findings.

```
* ENTER FINDINGS

** FEMALE
THE BEST MATCHES TO YOUR DESCRIPTION FOLLOW:
1) SEX = FEMALE

IS THIS OK? (Y/N): Y

** SERUM CALCIUM 13.5
THE BEST MATCHES TO YOUR DESCRIPTION FOLLOW:
1) TOTAL SERUM CALCIUM = 13.5 MG/100ML

IS THIS OK? (Y/N): Y

** COMA
THE BEST MATCHES TO YOUR DESCRIPTION FOLLOW:
1) LEVEL OF CONSCIOUSNESS = COMA

IS THIS OK? (Y/N): Y

** QUIT
```

At this point a total of three findings has been entered.

```
• ENTER HYPOTHESIS
```

The user now wants to enter an hypothesis.

```
•• PHPT

THE BEST MATCHES TO YOUR DESCRIPTION FOLLOW:
1) PRIMARY HYPERPARATHYROIDISM PRESENT

IS THIS OK? (Y/N): Y
```

NESTOR uses its synonym dictionary to interpret PHPT as primary hyperparathyroidism.

```
•• QUIT
```

This hypothesis will be called USER-HYP1.

The hypothesis PHPT now has a label, USER-HYP1, by which it can be referenced subsequently. Although this particular hypothesis consists of only one disease, in general an hypothesis can contain an arbitrary number of diseases.

```
• CRITIQUE USER-HYP1
```

The critiquing program module first explains the links known to NESTOR that connect the etiology of PHPT to the three findings in the current case. This orients the user to the general causal structure of the case. Next, a chart is displayed that gives the user a quantitative, probabilistic sense for how the likelihood of the USER-HYP1 hypothesis (PHPT) depends on the introduction of each finding. Overall, the critiquing task is meant to expose the strengths and weaknesses of a given hypothesis as a causal explanation of a given set of findings.

```
---------------------------------------------------------------------------
THE FOLLOWING CAUSAL MODEL EXPLAINS THE MECHANISMS BY WHICH THE
USER-HYP1 HYPOTHESIS COULD CAUSE THE FINDINGS:

 PRIMARY HYPERPARATHYROIDISM IS CHARACTERIZED BY THE PRESENCE OF
 INCREASED PTH. INCREASED PTH CAN RESULT IN AN INCREASED TOTAL SERUM
 CALCIUM. THIS THEN IS CAPABLE OF PRODUCING A DECREASED LEVEL OF
 CONSCIOUSNESS.
---------------------------------------------------------------------------
```

The above explanation was generated from NESTOR's top-level causal model[17] *A more detailed explanation using the bottom-level model will be shown later. The text is a direct English translation of the internal representation of this case.*[18]

P(USER-HYP1 | F1 TO Fn)

FINDINGS	NUMERIC FORM (PERCENT)	GRAPHIC FORM (PERCENT) 0 100
F1. NO FINDINGS AVAILABLE	.05 TO .1	*
F2. SEX IS FEMALE	+0 TO 1	*
F3. TOTAL SERUM CALCIUM IS 13.5	3 TO 42	*********
F4. LEVEL OF CONSCIOUSNESS IS COMA	0	*

The above chart communicates how each finding impacts on the score of the USER-HYP1 hypothesis[19]*. The chart orders the findings by placing the noncausal ones first (i.e., those not caused by some etiology of the diseases of the hypothesis being scored, such as sex), then the causal ones are ordered according to their causal interdependencies For example serum calcium appears before coma, because in PHPT it is the serum calcium that affects the patient's level of consciousness.*

The chart first shows that the prior probability of PHPT is between 0.5/1000 and 1/1000. When the finding female is introduced the probability of PHPT does not increase appreciably, which is logical since the sex of the patient contains little discriminatory information. However, when the serum calcium is considered, the potential probability of PHPT rises to 42%. Notice the wide range between a lower bound of 3% and an upper bound of 42%; this is reflected in the large number of asterisks on line F3 which visually communicates the wide probability bounds. Later we will see that this is attributable to the bounds of the probabilities in the knowledge-base. Finally, when coma is considered, the probability of PHPT drops to

---

[17] The relationship of this method to previous work by Wallis and Shortliffe [Wallis 82] and Patil [Patil 81] is discussed in Section 6.3.

[18] In order to avoid a stilted, mechanical generation of text, the conversion of a given internal causal representation to English randomly inserts syntactically different, semantically equivalent verb phrases. For this reason, later causal translations will produce slightly different syntactic versions that express the same semantics as the above paragraph.

[19] The relationship of this method to related work by Spiegelhalter and Knill-Jones [Spiegelhalter 84] is discussed in Section 6.3.

zero. This information plus the structure of the causal interaction printed earlier indicate that a serum calcium of 13.5 mg/100ml is inadequate to account for the coma observed in this patient. Something else must be happening.

The user will shortly ask NESTOR to suggest a better alternative to PHPT.

First, however, the user chooses to change the assumptions so that the probabilities in the knowledge base are set to the average value between their bounds[20] *. The user is interested in seeing the contribution that bounded probabilistic knowledge makes to the precision of the scores displayed in the previous critique chart.*

The user does, however, decide to retain the knowledge-based aggregation function; the choices for an aggregation function are: mathematical, knowledge-based, or independent. The aggregation function is the computational method used in calculating local joint conditional probabilities in a causally structured graph. The mathematical method is the most conservative since it relies totally on the axiomatic principles of probability to make conditional probability calculations. However, the price of this conservatism is wide bounds on the resulting probability. The independent method is the least conservative because it categorically assumes independence in all cases, and it yields very tight bounds on the resulting probability. The problem with this method is that the assumptions of independence may often be incorrect. Finally, the knowledge-based method attempts to use all available causal knowledge to guide the calculations so that the resulting probability is bounded reasonably tightly and yet is also valid.

---

[20] This assumes that the midpoint between the bounds is the expected value of the probability.

```
* SET ASSUMPTIONS

   ---- ASSUMPTION SET 1 ----
   AGGREGATION FUNCTION: KNOWLEDGE-BASED
   PROBABILISTIC DATA: BOUNDED

ENTER NEW VALUES? (Y/N): Y

   ---- ASSUMPTION SET 1 ----
** AGGREGATION FUNCTION: KNOWLEDGE-BASED
** AVERAGE BOUNDED PROBABILISTIC DATA? (Y/N): Y

   ---- ASSUMPTION SET 2 ----
** AGGREGATION FUNCTION: _
```

The user has entered a new assumption set. A null entry for the aggregation function of the second assumption set indicates that no additional assumption sets will be used; the use of an assumption set beyond the first set will be fully explained later in the context of generating hypotheses in Chapter 5. Briefly, additional assumption sets are used if previous ones did not totally order the hypothesis set, and a total ordering was requested by the user.

```
* CRITIQUE USER-HYP1

-------------------------------------------------------------------------------
THE FOLLOWING CAUSAL MODEL EXPLAINS THE MECHANISMS BY WHICH THE
USER-HYP1 HYPOTHESIS COULD CAUSE THE FINDINGS:

 PRIMARY HYPERPARATHYROIDISM IS CHARACTERIZED BY THE PRESENCE OF
 INCREASED PTH. INCREASED PTH CAN LEAD TO AN INCREASED TOTAL SERUM
 CALCIUM. THIS IN TURN IS CAPABLE OF PRODUCING A DECREASED LEVEL OF
 CONSCIOUSNESS.
-------------------------------------------------------------------------------


                                        P(USER-HYP1 | F1 TO Fn)

          FINDINGS                  NUMERIC FORM      GRAPHIC FORM
                                      (PERCENT)         (PERCENT)
                                                    0             100
-------------------------------------------------------------------------------
F1. NO FINDINGS AVAILABLE          |     .075      |*                |
F2. SEX IS FEMALE                  |    +0 TO 1    |*                |
F3. TOTAL SERUM CALCIUM IS 13.5    |    6 TO 16    |  ***            |
F4. LEVEL OF CONSCIOUSNESS IS COMA |       0       |*                |
-------------------------------------------------------------------------------
```

Notice that when the serum calcium is introduced the probability of PHPT is more

tightly bounded than before[21] *This indicates that at least in the current case a large portion of the original imprecision is due to the bounding of the probabilistic knowledge and not in the computational method. This is not the kind of check a typical user would make every time. However, the point is that the assumptions NESTOR uses are made readily visible to the user and can be easily altered.*

```
* SET ASSUMPTIONS

   ---- ASSUMPTION SET 1 ----
   AGGREGATION FUNCTION: KNOWLEDGE-BASED
   PROBABILISTIC DATA: AVERAGED

ENTER NEW VALUES? (Y/N): Y

   ---- ASSUMPTION SET 1 ----
** AGGREGATION FUNCTION: KNOWLEDGE-BASED
** AVERAGE BOUNDED PROBABILISTIC DATA? (Y/N): N

   ---- ASSUMPTION SET 2 ----
** AGGREGATION FUNCTION:
```

The above interaction shows the assumptions being set back to ***knowledge-based aggregation*** *and* ***bounded data.***

```
* GENERATE HYPOTHESIS
```

The user requests that NESTOR generate the most likely hypothesis to account for the original three findings.

---

[21] The bounds are not reduced to a single point probability because the term P(F) in P(H | F) = P(H & F) / P(F) is bounded rather calculated exactly. However, the value of P(H & F) *is* a point probability due to the assumption set that is being used. Section 6.2 discusses in detail the method used to bound P(F), and why this is necessary.

```
    RELATIVE SCORE      P(H | F)            HYPOTHESIS
-------------------------------------------------------------------------

1.   0 TO 100          0 TO 100          METASTATIC CANCER

-------------------------------------------------------------------------
```

The lack of precision in the relative score and P(H / F) results from NESTOR not devoting any time to calculating them; this is a time-saving default which can be changed by the user. Section 5.5.6 discusses how the user requests tighter bounds and section 6.2.3 discusses how NESTOR establishes them.

```
TIME TAKEN IN HYPOTHESIS FORMATION: 1.11 MINUTES
```

This is the CPU time required to locate the most likely hypothesis. This information, along with that directly below, is displayed each time NESTOR generates an hypothesis.

```
HYPOTHESES GENERATED: 32      HYPOTHESES SCORED: 16

ASSUMPTIONS USED:
     AGGREGATION: KNOWLEDGE-BASED      PROBABILISTIC DATA: BOUNDED

THE TOP-RANKED HYPOTHESIS WILL BE CALLED NESTOR-HYP1.
```

Out of a total of $2^7 = 128$ possible hypotheses, NESTOR only had to consider 32 and of these only score 16 in order to guarantee that metastatic cancer is the most probable hypothesis[22] *This savings is due to the branch and bound search technique used. Section 5.7.3.4 contains a general discussion of the amount of savings in search time that can be expected when using this technique.*

The interpretation of the relative score and P(H / F) will be explained later.

This newly generated hypothesis, metastatic cancer, is given the label NESTOR-HYP1.

---

[22] It is important to realize that metastatic cancer is the most likely hypothesis relative to any hypothesis that consists of a subset of the seven diseases that NESTOR *currently* represents. In particular, in a larger domain there would be other causes of coma that are more probable than metastatic cancer and therefore some other hypothesis might account for the findings better than metastatic cancer. This emphasizes the importance of interpreting the results of NESTOR's hypothesis generation procedure in the context of the current knowledge-base. Section 6.2 discusses a method of calculating the posterior probability of such an hypothesis, which does not assume an hypothesis space that is limited to the combinations of currently represented diseases. Thus, although an hypothesis can not be generated from diseases not currently represented, at least the probability of the *most likely* hypothesis which *is* generated will not be over-estimated.

• COMPARE USER-HYP1 NESTOR-HYP1

The COMPARE task first displays the case-specific causal model of each hypothesis. Then, it quantitatively compares how well the two hypotheses explain the findings.[23]

```
--------------------------------------------------------------------------------
THE FOLLOWING CAUSAL MODEL EXPLAINS THE MECHANISMS BY WHICH THE
USER-HYP1 HYPOTHESIS COULD CAUSE THE FINDINGS:

 PRIMARY HYPERPARATHYROIDISM IS CHARACTERIZED BY THE PRESENCE OF
 INCREASED PTH. INCREASED PTH IS CAPABLE OF CAUSING AN INCREASED
 TOTAL SERUM CALCIUM. THIS IN TURN IS ABLE TO CAUSE A DECREASED
 LEVEL OF CONSCIOUSNESS.

--------------------------------------------------------------------------------
THE FOLLOWING CAUSAL MODEL EXPLAINS THE MECHANISMS BY WHICH THE
NESTOR-HYP1 HYPOTHESIS COULD CAUSE THE FINDINGS:

 METASTATIC CANCER IS CHARACTERIZED BY THE PRESENCE OF METASTATIC CANCER
 CELLS. METASTATIC CANCER CELLS CAN RESULT IN AN INCREASED TOTAL SERUM
 CALCIUM, AND TUMOR FORMATION IN THE BRAIN. TUMOR FORMATION IN THE BRAIN
 AND INCREASED TOTAL SERUM CALCIUM ARE CAPABLE OF PRODUCING A DECREASED
 LEVEL OF CONSCIOUSNESS.

--------------------------------------------------------------------------------
```

Notice the critical difference between the USER-HYP1 and the NESTOR-HYP1 hypothesis: the NESTOR-HYP1 hypothesis can cause coma by way of a brain tumor.

---

[23] The current COMPARE command is viewed as only one step in the development of comparative explanation techniques in the field of computer-aided medical decision making. Sections 6.4.2 and 6.4.3 discuss future extensions to the COMPARE command by which more explicit and detailed comparisons can be provided.

```
THE TABLE BELOW CONTAINS THE PROBABILITY OF THE USER-HYP1 HYPOTHESIS
UNDER THE ASSUMPTION THAT IT AND THE NESTOR-HYP1 HYPOTHESIS ARE THE
ONLY HYPOTHESES POSSIBLE.
NOTE: UNDER THIS ASSUMPTION THE PROBABILITY OF THE NESTOR-HYP1 HYPOTHESIS
IS JUST 100 MINUS THE PROBABILITY OF THE USER-HYP1 HYPOTHESIS.
```

This explains the method that NESTOR uses to limit the comparison to just two hypotheses. Section 6.1.2 contains a detailed discussion of how this method of quantitatively comparing two hypotheses is equivalent (in terms of information content) to comparing the ratio of the posterior probabilities of the two hypotheses.

P(USER-HYP1 | F1 TO Fn)

FINDINGS	NUMERIC FORM (PERCENT)	GRAPHIC FORM (PERCENT) 0 100
F1. NO FINDINGS AVAILABLE	4 TO 17	****
F2. SEX IS FEMALE	6 TO 21	****
F3. TOTAL SERUM CALCIUM IS 13.5	5 TO 47	*********
F4. LEVEL OF CONSCIOUSNESS IS COMA	0	*

The prior probability of the USER-HYP1 (PHPT) hypothesis is shown as F1. Since the prior probability of metastatic cancer is larger than that of PHPT, the prior likelihood of PHPT is low. The sex of patient being female slightly favors PHPT. The effect of a serum calcium of 13.5 is difficult to judge, since the bounds are so wide. However, the introduction of coma clearly indicates that metastatic cancer is the most likely hypothesis. Thus, the ability of metastatic cancer to cause coma by way of a tumor was crucial in it being more likely than PHPT.

In the next case the patient has no coma, but she is known to have muscle weakness and constipation. The user again believes that PHPT may be the cause, and would like to know the details of how it can causally account for these findings.

```
* DISPLAY FINDINGS

1) SEX = FEMALE
2) TOTAL SERUM CALCIUM = 13.5 MG/100ML
3) GENERAL MUSCLE STRENGTH = WEAK
4) CONSTIPATION = PRESENT

* CRITIQUE USER-HYP1, OPTIONS
```

The request for OPTIONS allows the user to control several aspects of the critiquing task.

```
** DO YOU WANT AN ABBREVIATED (A)  OR A DETAILED (D)  DISPLAY
```

```
   OF THE CAUSAL MODEL THAT RELATES THE DISEASE ETIOLOGY TO THE
   FINDINGS (A/D): D
```

The user wants NESTOR to give a detailed explanation of how PHPT can cause the findings. NESTOR will use its detailed causal model to do this.

```
** DO YOU WANT AN INCREMENTAL (I)  FINDING-BY-FINDING  PROBABILITY
   ASSESSMENT OR JUST THE FINAL (F)  PROBABILITY GIVEN ALL THE
   FINDINGS (I/F):  I
```

The incremental option was used by default in the previous examples and is again chosen here. It requires more computation, but the results give the user a greater sense for the strengths and weaknesses of the hypothesis in accounting for the findings.

```
** RESTRICT THE DISEASE SET? (Y/N): N
```

The user does not wish to restrict the set of diseases that are used in generating competing hypotheses against which the USER-HYPI hypothesis is implicitly being compared during the critiquing task.

```
** INCLUDE USER-SELECTED DISEASES IN ALL HYPOTHESES
   GENERATED? (Y/N): N
```

Including diseases would force every alternative to the USER-HYPI hypothesis to contain specific diseases which the user could specify. This option is more often used in hypothesis generation when the user has a strong notion that the patient has certain diseases and wants these included in every hypothesis.

```
** EXCLUDE USER-SELECTED DISEASES IN ALL HYPOTHESES
   GENERATED? (Y/N): Y
      1. PHEOCHROMOCYTOMA
      2. SARCOIDOSIS
      3. GRAVES DISEASE
      4. NON-METASTATIC CANCER
      5. METASTATIC CANCER
      6. MYELOMA
      7. PRIMARY HYPERPARATHYROIDISM
   CHOICE(S): 4 5 6
```

The user believes that the patient does not have any form of cancer and wants the USER-HYPI hypothesis (PHPT) to be critiqued only with respect to hypotheses that contain noncancerous diseases.

```
** LIMIT THE NUMBER OF DISEASES IN AN HYPOTHESIS TO NO MORE
   THAN (#/ALL): 1
```

This specifies that the USER-HYP1 hypothesis is to be critiqued only against hypotheses that contain a single disease. Apparently the user is confident that there is only one etiological basis for the current set of findings.

```
** THE GOAL FOR RANGE OF THE BOUNDS ON THE POSTERIOR PROBABILITY
   OF THE USER-HYP1 HYPOTHESIS IS (IN PERCENT): 10
```

The smaller this number, the more computation NESTOR must perform. It may not always be possible to attain the goal of a range of 10% in the final posterior probability, but NESTOR will attempt to satisfy it within the time limits set forth.

```
** LIMIT THE REAL-TIME MINUTES OF HYPOTHESIS GENERATION TO: 5
```

Here the user indicates that in no case should NESTOR compute for more than five minutes before rendering a critique; this is the maximum amount of time that the user is willing to wait for a result. The use of bounds allows NESTOR to return the tightest result it has calculated thus far, even if this does not meet the goal of a 10% range on the bounds as set above. Thus, the search will terminate after either the passing of 5 real-time minutes of computation or the achievement of bounds with a range of less than 10% on the posterior probability, whichever comes first.

```
--------------------------------------------------------------------------------
THE FOLLOWING CAUSAL MODEL EXPLAINS THE MECHANISMS BY WHICH THE
USER-HYP1 HYPOTHESIS COULD CAUSE THE FINDINGS:

 PRIMARY HYPERPARATHYROIDISM IS CHARACTERIZED BY THE PRESENCE OF INCREASED
 PTH. INCREASED PTH CAN CAUSE AN INCREASED RENAL RESORPTION OF CALCIUM,
 AN INCREASED GASTROINTESTINAL ABSORPTION OF CALCIUM, AND AN INCREASED
 OSTEOCLAST ACTIVITY. INCREASED OSTEOCLAST ACTIVITY CAN RESULT IN AN INCREASED
 RESORPTION OF BONE. INCREASED RENAL RESORPTION OF CALCIUM AND INCREASED
 GASTROINTESTINAL ABSORPTION OF CALCIUM AND INCREASED RESORPTION OF BONE
 ARE CAPABLE OF PRODUCING AN INCREASED TOTAL SERUM CALCIUM. INCREASED TOTAL
 SERUM CALCIUM CAN CAUSE A DECREASED NEURONAL AND NEUROMUSCULAR FUNCTION.
 DECREASED NEURONAL AND NEUROMUSCULAR FUNCTION CAN PRODUCE A DECREASED GENERAL
 MUSCLE STRENGTH, AND A DECREASED GI TRACT MOTILITY. DECREASED GI TRACT
 MOTILITY CAN PRODUCE CONSTIPATION.
--------------------------------------------------------------------------------
```

It is apparent that this causal explanation contains much more detail than the previous ones. From it the user can understand many more of the subtleties of the causal interactions. For example, the effect of the serum calcium on muscle weakness and constipation share the common mechanism of a depressed neuronal and neuromuscular function; in a sense

constipation results from weak gastrointestinal muscles[24] *. The user may not always wish to be confronted with such detailed mechanisms, but it is good to be able to provide them when they are desired.*

P(USER-HYP1 | F1 TO Fn)

FINDINGS	NUMERIC FORM (PERCENT)	GRAPHIC FORM (PERCENT) 0 100
F1. NO FINDINGS AVAILABLE	.075	*
F2. SEX IS FEMALE	+0 TO 1	*
F3. TOTAL SERUM CALCIUM IS 13.5	99 TO 100	*
F4. CONSTIPATION IS PRESENT	99 TO 100	*
F5. GENERAL MUSCLE STRENGTH IS WEAK	97 TO 100	*

The above chart shows that PHPT is by far the most likely diagnosis among its contenders[25] *. Recall that its contenders are single disease, noncancerous hypotheses, namely: pheochromocytoma, sarcoidosis, and Graves' disease. The chart indicates that none of these diseases is capable of accounting for a hypercalcemia as high as 13.5 mg/100ml. Since PHPT* ***can*** *account for a serum calcium of 13.5, it has a posterior probability of 100% relative to its currently defined competitors.*

Now the user decides to consider a different case and enters three new findings (the actual entry is not shown) and then displays them.

```
* DISPLAY FINDINGS

1) WEIGHT = DECREASED
2) ANEMIA = PRESENT
3) TOTAL SERUM CALCIUM = 11.7 MG/100ML
```

Next, NESTOR is asked to generate the most likely hypothesis to account for these three findings.

---

[24] Although it is possible that other mechanisms may also cause these two findings, the purpose of this example is simply to show that a shared mechanism can be used to explain several findings.

[25] The chart was created using the averages of bounded probabilities.

```
* GENERATE HYPOTHESIS

    RELATIVE SCORE      P(H | F)            HYPOTHESIS
-------------------------------------------------------------------------

1.   31 TO 100          3 TO 96            METASTATIC CANCER

-------------------------------------------------------------------------

TIME TAKEN IN HYPOTHESIS FORMATION: 0.49 MINUTES

HYPOTHESES GENERATED: 14      HYPOTHESES SCORED: 6

ASSUMPTIONS USED:
     AGGREGATION: KNOWLEDGE-BASED      PROBABILISTIC DATA: BOUNDED

THE TOP-RANKED HYPOTHESIS WILL BE CALLED NESTOR-HYP2.
```

*The **relative score** always assigns the upper bound of the most probable hypothesis to be 100. The lower bound is set relative to this. Thus, in the above case the 31 means that the lower bound of the probability of metastatic cancer is 31% as great as its upper bound probability. The posterior probability of metastatic cancer is between 3% and 96%. This is a wide range because by default NESTOR does not try to do any more calculations than are necessary to determine the most likely hypothesis; the range 3 to 96 is just a side effect of this process. If the user desired tighter bounds, then this could be explicitly requested. Alternatively, the user may CRITIQUE the hypothesis in order to tighten the bounds on the posterior probability.*

A number of hypotheses can potentially account for the three findings in this case, as for example myeloma, or some combination of metastatic cancer and another disease. Metastatic cancer was selected over other possibilities primarily on the basis of its higher prior probability.

```
* GENERATE HYPOTHESIS, OPTIONS
```

The hypothesis generation task allows the user to override defaults by requesting options.

```
** TO FIND THE N MOST LIKELY HYPOTHESES ENTER N: 2
```

The user decides that it would be useful to know the next most probable hypothesis after metastatic cancer, so a request is made to have the top two hypotheses generated. Theoretically, up to 2^N top hypotheses can be generated, where N is the number of diseases in the knowledge-base (currently 7); typically, however, no more than 10 of the top hypotheses are of interest.

```
** DO YOU WISH TO SEPARATE THE OVERLAP OF THE PROBABILITY RANGES
   OF THE TOP TWO HYPOTHESES BY USING SUCCESSIVELY MORE RADICAL
   ASSUMPTIONS IF NECESSARY (NOTE: THIS MAY TAKE LONGER)?
   (Y/N): N
```

The above option will be explained in the next example.

```
** RESTRICT THE DISEASE SET? (Y/N): N
** INCLUDE USER-SELECTED DISEASES IN ALL HYPOTHESES
   GENERATED? (Y/N): N
** EXCLUDE USER-SELECTED DISEASES IN ALL HYPOTHESES
   GENERATED? (Y/N): N
** LIMIT THE NUMBER OF DISEASES IN AN HYPOTHESIS TO NO MORE
   THAN (#/ALL): ALL
** THE GOAL FOR THE RANGE OF THE BOUNDS ON THE POSTERIOR
   PROBABILITY OF THE MOST LIKELY HYPOTHESIS
   IS (IN PERCENT):  100
```

The 100 indicates that the bounds can be as large as 0 to 100; it is clear from this that the user is not interested in NESTOR spending any time narrowing these probability bounds.

```
** LIMIT THE REAL-TIME MINUTES OF HYPOTHESIS GENERATION TO: 5
```

Here the user states a desire to wait no more than five minutes on the result. If after this time NESTOR has not found the two hypotheses that are guaranteed to be the most probable two, it will simply inform the user of the most likely two it has generated thus far.

```
    RELATIVE SCORE      P(H | F)            HYPOTHESIS
-------------------------------------------------------------------------

1.   31 TO 100          3 TO 96          METASTATIC CANCER

2.   3 TO 12            0 TO 28          NON-METASTATIC CANCER

-------------------------------------------------------------------------

TIME TAKEN IN HYPOTHESIS FORMATION: 0.52 MINUTES

HYPOTHESES GENERATED: 14     HYPOTHESES SCORED: 6

ASSUMPTIONS USED:
     AGGREGATION: KNOWLEDGE-BASED     PROBABILISTIC DATA: BOUNDED

THE TOP-RANKED HYPOTHESIS WILL BE CALLED NESTOR-HYP3
```

Non-metastatic cancer was found to be the second most likely hypothesis to account for anemia, weight loss, and a serum calcium of 11.7 mg/100ml. The upper of bound of the relative score of non-metastatic cancer is 12, indicating that it is 12% as great as the upper bound of the posterior probability of metastatic cancer. Similarly, its lower bound is 3% as

great as the upper bound of of the posterior probability of metastatic cancer. The relative scores allow the user to see immediately that metastatic cancer is at best 100/3 = 33.3 times as great as non-metastatic cancer, and at worst 31/12 = 2.6 times as great.

Chapter 4 describes the scoring metric used to rank order hypotheses. Chapter 5 presents a search method which uses the scoring metric to locate the N most likely hypotheses. Section 6.1.2 discusses the information that is used to calculate relative scores.

Next, the user adds two more findings (the entry of the findings is not shown) and displays all five findings.

```
* DISPLAY FINDINGS

1) WEIGHT = DECREASED
2) ANEMIA = PRESENT
3) TOTAL SERUM CALCIUM = 11.7 MG/100ML
4) SERUM PHOSPHATE = DECREASED
5) SERUM CHLORIDE = INCREASED
```

The user also adds a second assumption set (entry of the assumption sets are not shown) and displays the two assumption sets that are currently in effect.

```
* DISPLAY ASSUMPTIONS

   ---- ASSUMPTION SET 1 ----
   AGGREGATION FUNCTION: KNOWLEDGE-BASED
   PROBABILISTIC DATA: BOUNDED

   ---- ASSUMPTION SET 2 ----
   AGGREGATION FUNCTION: KNOWLEDGE-BASED
   PROBABILISTIC DATA: AVERAGED

* GENERATE HYPOTHESIS, OPTIONS

** TO FIND THE N MOST LIKELY HYPOTHESES ENTER N: 1
** DO YOU WISH TO SEPARATE THE OVERLAP OF THE PROBABILITY RANGES
   OF THE TOP TWO HYPOTHESES BY USING SUCCESSIVELY MORE RADICAL
   ASSUMPTIONS IF NECESSARY (NOTE: THIS MAY TAKE LONGER)?
   (Y/N): Y
```

By requesting the above option the user indicates that NESTOR is to first use assumption set 1 in order to locate the hypothesis with the greatest upper bound probability. If this hypothesis has a lower bound which is less than the upper bound of another hypothesis, then the second assumption set is to be used. Typically, the higher the order of an assumption set the more radical its assumptions, which then leads to tighter bounds on the resulting probability scores. This increases the chances that the probability bounds of the top hypothesis

will not overlap that of other hypotheses. In any case, the user is always informed of the assumption set that was required to finally separate the top hypothesis from all the others.

If the user does not use this option, NESTOR will use only the first assumption set in order to find the hypothesis with the greatest upper bound probability.

```
** RESTRICT THE DISEASE SET? (Y/N): N
** INCLUDE USER-SELECTED DISEASES IN ALL HYPOTHESES
   GENERATED? (Y/N): N
** EXCLUDE USER-SELECTED DISEASES IN ALL HYPOTHESES
   GENERATED? (Y/N): N
** LIMIT THE NUMBER OF DISEASES IN AN HYPOTHESIS TO NO MORE
   THAN (#/ALL): ALL
** THE GOAL FOR THE RANGE OF THE BOUNDS ON THE POSTERIOR
   PROBABILITY OF THE MOST LIKELY HYPOTHESIS
   IS (IN PERCENT):  100
** LIMIT THE REAL-TIME MINUTES OF HYPOTHESIS GENERATION TO: 5

    RELATIVE SCORE      P(H | F)            HYPOTHESIS
-------------------------------------------------------------------------

1.   16 TO 100          0 TO 77          METASTATIC CANCER
                                          PRIMARY HYPERPARATHYROIDISM

-------------------------------------------------------------------------

TIME TAKEN IN HYPOTHESIS FORMATION: 4.25 MINUTES

HYPOTHESES GENERATED: 80     HYPOTHESES SCORED: 42

ASSUMPTIONS USED:
     AGGREGATION: KNOWLEDGE-BASED     PROBABILISTIC DATA: AVERAGED

THE TOP-RANKED HYPOTHESIS WILL BE CALLED NESTOR-HYP4
```

Note that the second assumption set was required in order to separate the probability bounds of the top hypothesis from other contenders. The difference between the first and the second assumption set is that the first set uses bounded probabilistic knowledge whereas the second averages those probabilities before using them.

Here NESTOR has generated a multiple disease hypothesis to account for the findings. Apparently, PHPT is being used to account for the low serum phosphate and the high serum chloride values.

The user is curious about whether there is a single disease that can account for all the

findings, and if so then how well does it compare to the NESTOR-HYP4 hypothesis just generated. Thus, the following request is made.

```
* GENERATE HYPOTHESIS, OPTIONS

** TO FIND THE N MOST LIKELY HYPOTHESES ENTER N: 1
** DO YOU WISH TO SEPARATE THE OVERLAP OF THE PROBABILITY RANGES
   OF THE TOP TWO HYPOTHESES BY USING SUCCESSIVELY MORE RADICAL
   ASSUMPTIONS IF NECESSARY (NOTE: THIS MAY TAKE LONGER)?
   (Y/N): Y
** RESTRICT THE DISEASE SET? (Y/N): N
** INCLUDE USER-SELECTED DISEASES IN ALL HYPOTHESES
   GENERATED? (Y/N): N
** EXCLUDE USER-SELECTED DISEASES IN ALL HYPOTHESES
   GENERATED? (Y/N): N
** LIMIT THE NUMBER OF DISEASES IN AN HYPOTHESIS TO NO MORE
   THAN (#/ALL): 1
```

The above option indicates that the user is seeking a single disease hypothesis to account for the findings.

```
** THE GOAL FOR THE RANGE OF THE BOUNDS ON POSTERIOR PROBABILITY
   OF THE MOST LIKELY HYPOTHESIS IS (IN PERCENT):  100
** LIMIT THE REAL-TIME MINUTES OF HYPOTHESIS GENERATION TO: 5

    RELATIVE SCORE      P(H | F)              HYPOTHESIS
-------------------------------------------------------------------------


1.   94 TO 100          50 TO 52     PRIMARY HYPERPARATHYROIDISM

-------------------------------------------------------------------------

TIME TAKEN IN HYPOTHESIS FORMATION: .82 MINUTES

HYPOTHESES GENERATED: 9      HYPOTHESES SCORED: 8

ASSUMPTIONS USED:
     AGGREGATION: KNOWLEDGE-BASED      PROBABILISTIC DATA: AVERAGED

THE TOP-RANKED HYPOTHESIS WILL BE CALLED NESTOR-HYP5
```

PHPT is the most likely single disease hypothesis. Notice that the posterior probability is 50% to 52%, which seems inconsistent with PHPT not being the most likely hypothesis. However, these bounds are with respect to the restricted set of all ***single*** *disease hypotheses, not all* ***possible*** *hypotheses, which would include the hypothesis metastatic cancer + PHPT, currently the best hypothesis overall.*

• COMPARE NESTOR-HYP4 NESTOR-HYP5

In order to judge the relative strengths and weaknesses of metastatic cancer + PHPT (NESTOR-HYP4) vs. PHPT alone (NESTOR-HYP5), the user has asked NESTOR to compare them.

THE FOLLOWING CAUSAL MODEL EXPLAINS THE MECHANISMS BY WHICH THE NESTOR-HYP4 HYPOTHESIS COULD CAUSE THE FINDINGS:

METASTATIC CANCER IS CHARACTERIZED BY THE PRESENCE OF METASTATIC CANCER CELLS. PRIMARY HYPERPARATHYROIDISM IS CHARACTERIZED BY THE PRESENCE OF INCREASED PTH. METASTATIC CANCER CELLS AND INCREASED PTH CAN PRODUCE AN INCREASED TOTAL SERUM CALCIUM. INCREASED PTH CAN LEAD TO A DECREASED SERUM PHOSPHATE, AND AN INCREASED SERUM CHLORIDE. METASTATIC CANCER CELLS CAN PRODUCE A DECREASED WEIGHT, AND ANEMIA.

The above explanation confirms our expectations that metastatic cancer is being used to account for the anemia and weight loss, while PHPT is accounting for the low serum phosphate and the high serum chloride. Both diseases contribute to the hypercalcemia.

THE FOLLOWING CAUSAL MODEL EXPLAINS THE MECHANISMS BY WHICH THE NESTOR-HYP5 HYPOTHESIS COULD CAUSE THE FINDINGS:

PRIMARY HYPERPARATHYROIDISM IS CHARACTERIZED BY THE PRESENCE OF INCREASED PTH. INCREASED PTH IS CAPABLE OF CAUSING AN INCREASED TOTAL SERUM CALCIUM, A DECREASED SERUM PHOSPHATE, AND AN INCREASED SERUM CHLORIDE. NORMAL PHYSIOLOGY CAN PRODUCE A DECREASED WEIGHT, AND ANEMIA.

The above explanation indicates that PHPT is again accounting for the low serum phosphate and the high serum chloride. Additionally, it is being considered to be the sole cause of the hypercalcemia. However, a major difference between this hypothesis and the NESTOR-HYP4 hypothesis is that this one uses normal physiology to explain the weight decrease and anemia. NESTOR knows that it is possible for people to lose weight (e.g., dieting) and to be anemic (e.g., a menstruating female) without any underlying pathology.

The following chart shows how the two hypotheses compare quantitatively.

THE TABLE BELOW CONTAINS THE PROBABILITY OF THE NESTOR-HYP4 HYPOTHESIS UNDER THE ASSUMPTION THAT IT AND THE NESTOR-HYP5 HYPOTHESIS ARE THE ONLY HYPOTHESES POSSIBLE.
NOTE: UNDER THIS ASSUMPTION THE PROBABILITY OF THE NESTOR-HYP5 HYPOTHESIS IS JUST 100 MINUS THE PROBABILITY OF THE NESTOR-HYP4 HYPOTHESIS.

P(NESTOR-HYP4 | F1 TO Fn)

FINDING	NUMERIC FORM (PERCENT)	GRAPHIC FORM (PERCENT) 0 100
F1. NO FINDINGS AVAILABLE	.7444	*
F2. SERUM PHOSPHATE IS DECREASED	+0 TO 1	*
F3. WEIGHT IS DECREASED	8 TO 9	*
F4. TOTAL SERUM CALCIUM IS 11.7	6 TO 11	**
F5. ANEMIA IS PRESENT	63 TO 75	***
F6. SERUM CHLORIDE IS INCREASED	63 TO 75	***

The prior probability, F1, shows that the two disease hypothesis NESTOR-HYP4 (consisting of metastatic cancer + PHPT) is much less likely a priori than the single disease hypothesis NESTOR-HYP5 (consisting of PHPT). The consideration of a low phosphate does not affect the relative likelihood of the two hypotheses, since PHPT accounts for it equally well in both. The introduction of weight loss does however slightly favor the two disease hypothesis which contains metastatic cancer as its cause. The serum calcium is explained about equally well by both hypotheses. However, when anemia is considered the probability of the two disease hypothesis increases considerably. This is because the likelihood of anemia given metastatic cancer is considerably greater than the likelihood of an anemia given that there is no underlying pathological process. Finally, the introduction of the serum chloride does not further change the relative likelihood of the two hypotheses since it is explained equally well by the PHPT in both hypotheses. In summary, the two disease hypothesis began as much less likely a priori than the single disease one, but after the introduction of a weight loss and anemia it became more likely.

• QUIT

Chapter 3

Knowledge Representation

The principle knowledge structure in NESTOR is an acyclic, causal graph. Each disease is represented as a node which contains a set of nodes and links. Non-disease nodes are used to represent etiologies, findings, and intermediate states or processes. Links are used to represent various aspects of the causal connection between nodes. The general form of this representation is known as a semantic network in artificial intelligence (AI), and has been used extensively in AI research [Barr 81]. As an example of a node and link structure in NESTOR, Figure 3-1 shows a portion of the graph representing the disease primary hyperparathyroidism (PHPT).

PTH is an etiological node, serum calcium and muscle strength are finding nodes, and neuromuscular function is an intermediate causal node. The links define the possible causal interactions of nodes, the causal direction of these interactions, and the probabilistic relationships between the cause and effect nodes.

NESTOR currently contains about 100 nodes and 200 links representing 7 diseases which cause hypercalcemia. This chapter explains in detail the acquisition and representation of nodes and links.

3.1. Nodes

The basic elements of a node are demonstrated by the example in Figure 3-2 for serum calcium.

The decomposition of a node into a primary object (or process), a list of constant properties, and a variable property has several advantages over just representing a node by its name. As will discussed in Section 3.1.2, the use of this representation facilitates the entry

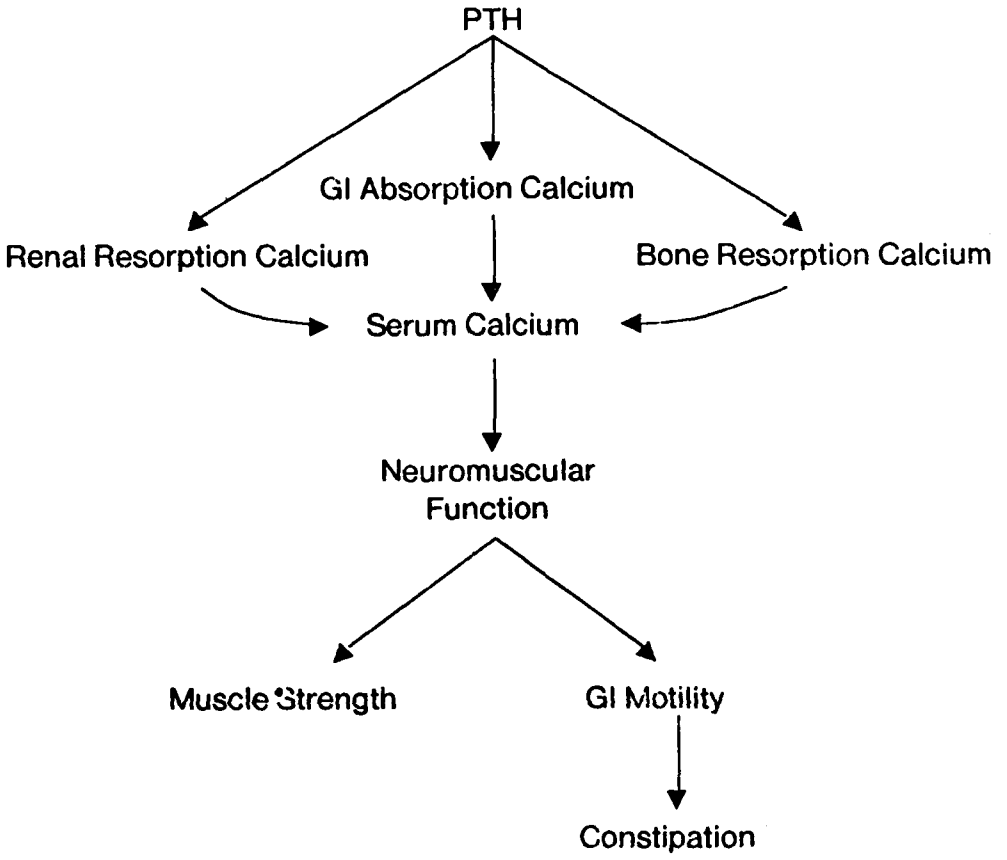


Figure 3-1: A Partial Causal Graph of PHPT

of findings because NESTOR can narrow an underspecified finding to a single one by asking for the value of discriminating properties, such as its location. Also, distinguishing the constant properties of a node from its variable property is a means of separating the contextual information of an object/process from its actual measured property. Currently, this is used as a means of focusing NESTOR's attention toward reasoning about the value of the variable property. In the future this information could be extended toward use in reasoning about how findings are related. For example knowing that one finding has the same location as another could be useful in inferring the possible location of the disease process. The possible values of the variable are also shown in Figure 3-2. This list actually represents a hierarchy of values as is shown in Figure 3-3.

```
Name:                  Total Serum Calcium
Object:                Calcium
Constant Properties:   Location: Blood
Variable Property:     Level
Variable Type:         Continuous
Variable Units:        mg/100ml
Variable Values:       ((decreased (0 9))
                        (normal (9 10.5))
                        (mildly-increased (10.5 12))
                        (moderately-increased (12 15))
                        (severely-increased (15 30))
                        (increased (10.5 30)))
```

Figure 3-2: An Example of a Finding Node

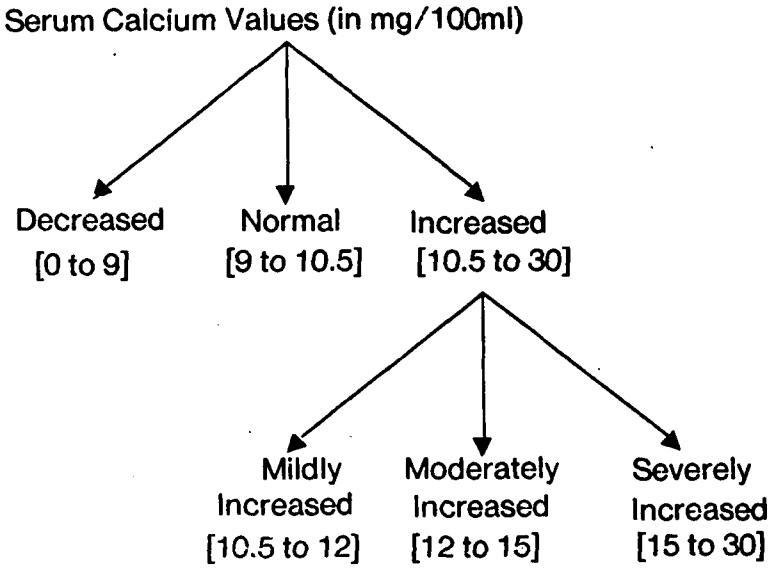


Figure 3-3: An Example of a Hierarchy of Node Values

NESTOR allows the specification of any connected range of values for a given node. Typically this capability is used to specify that a finding has a value within some range such

as *increased*. There are 3 variable types in NESTOR: binary, discrete, and continuous. Total serum calcium, the example in Figure 3-2, is a continuous variable. Note that even in the case of a continuous variable the values of the variable are divided into discrete intervals (such as 10.5 to 12 in the example). The finest granularity of interval values for a variable are assumed to be small enough so that different values within a given interval are not clinically significant from a diagnostic viewpoint. By discretizing continuous variables NESTOR is able to uniformly reason with only discrete valued variables, which simplifies the scoring procedure.

3.1.1. Disease Representation

A disease is represented as a node containing all the information listed in Figure 3-2. However, it also contains some additional information. Figure 3-4 shows an how the disease PHPT is represented.

```
Name: PHPT
Process:              Over-production-of-PTH
Constant Properties:  Anatomical-origin: Parathyroid-gland
Variable Property:    Existence
Variable Type:        Binary
Variable Units:       none
Variable Values:      ((present)(absent))
Prior Probability:    (0.05% to 0.1%)
Etiology Nodes:       (PTH = increased)
Findings:             (Serum-calcium, Serum-phosphate, ...)
```

Figure 3-4: An Example of a Disease Node

There are 3 added pieces of information: the prior probability of the disease, the etiological node(s) of the disease, and a list of the finding variables associated with the disease. The finding variables are shown symbolically in Figure 3-4, but are numerical pointers in the actual internal representation. These pointers allow NESTOR to determine if a given set of findings can possibly be explained by a particular disease.

Currently, seven causes of hypercalcemia are represented in NESTOR's knowledge-base, namely, primary hyperparathyroidism, metastatic cancer, non-metastatic cancer,

myeloma, Graves' disease, sarcoidosis, and pheochromocytoma. There are about ten findings per disease. This is relatively small due to the findings being multivalued rather than binary, and because only the most common findings in the diseases are represented. While the current knowledge-base is not large, it has been sufficient as an initial test of the basic principles underlying NESTOR's design.

3.1.2. Finding Representation and Entry

The findings of a patient are represented as nodes that have all the components of a basic node as discussed in above. Currently findings are assumed to exist with certainty and the user can not qualify them with a probability.

The procedure for acquiring findings from the user is based on an interactive discrimination net approach. There is a great deal of flexibility in how findings may be entered. To NESTOR a finding is described by a set of property-value pairs. For example if the finding INCREASED SERUM CALCIUM is entered, NESTOR will internally translate this to OBJECT: CALCIUM LOCATION: SERUM LEVEL: INCREASED. Either of these two forms is acceptable as input, but usually the user will enter the former. The examples that follow will show some of the range of NESTOR's flexibility in accepting findings. Then, the general algorithm will be explained.

3.1.2.1. Examples of Entering Findings

The user's input is underlined in all the examples that follow.

```
** INCREASED CALCIUM

WHICH OF THE FOLLOWING VALUES ARE PERMISSIBLE
   FOR THE PROPERTY: LOCATION ?
     1) BLOOD
     2) URINE
     3) LOCATION is not a necessary property to be considered
YOUR CHOICES: 1

THE BEST MATCH TO YOUR DESCRIPTION FOLLOWS:
1) TOTAL SERUM CALCIUM = INCREASED

IS THIS OK? (Y/N): Y
```

Here the user has specified increased calcium as the finding. However, this is

underspecified since NESTOR knows of two findings that fit that description, namely increased serum calcium and increased urinary calcium. Since location is the property that discriminates these two, NESTOR asks the user which location is intended.

In the next example the same finding is entered, but in this case the value 11 is supplied instead of the word *increased.*

```
** CALCIUM 11

THE BEST MATCH TO YOUR DESCRIPTION FOLLOWS:
1) TOTAL SERUM CALCIUM = 11 MG/100ML

IS THIS OK? (Y/N): Y
```

Since the permissible bounds of the serum calcium level include 11 (mg%), but those of urinary calcium do not, NESTOR is able to determine which finding is intended.

In the final example, the same finding is again being entered, but this time the initial input by the user is a blank line. NESTOR then proceeds to request the value of properties from the user.

```
** _

WHICH OF THE FOLLOWING VALUES ARE PERMISSIBLE
   FOR THE PROPERTY: OBJECT ?
     1) ACID-PHOSPHATASE
     2) ALKALINE-PHOSPHATASE
     3) BICARBONATE
     4) BLOOD
     5) BONE
     6) BRAIN
     7) CALCIUM
     8) CHLORIDE
     9) KIDNEY
     10) MAGNESIUM
     11) MUSCLE
     12) PARATHORMONE
     13) PHOSPHATE
     14) POTASSIUM
     15) RIB
     16) SHOULDER
     17) SKIN
     18) SKULL
     19) SPINE
     20) URINE
     21) OBJECT is not a necessary property to be considered
YOUR CHOICES: 7

WHICH OF THE FOLLOWING VALUES ARE PERMISSIBLE
   FOR THE PROPERTY: LOCATION ?
     1) BLOOD
     2) URINE
     3) LOCATION is not a necessary property to be considered
YOUR CHOICES: 1

THE BEST MATCH TO YOUR DESCRIPTION FOLLOWS:

1) TOTAL SERUM CALCIUM = ?

IS THIS OK? (Y/N): Y
PLEASE ENTER A VALUE FOR THIS FINDING.
POSSIBLE VALUES ARE:
  DECREASED: 0 TO 9
  NORMAL: 9 TO 10.5
  INCREASED: 10.5 TO 30
YOUR CHOICE: INCREASED
```

The properties are ordered so that the more basic ones are asked first. This example demonstrates that users can begin with no knowledge of NESTOR's specific encoding of findings, and then gradually learn its vocabulary as they answer discriminating questions. It is easy to underestimate the importance of such a feature until one has to deal with a system

that does not have it. In addition to its use in entering findings, it is also quite helpful to the knowledge engineer or expert who needs to specify the cause and effect nodes when defining a causal link (see Section 3.2.2).

3.1.2.2. The General Algorithm for Entering Findings

The entry algorithm receives a set of finding values or property-value pairs. It processes this in a 5 step process. The example of entering the finding HIGH CALCIUM will be used to demonstrate the effects of each step.

1. **Canonicalized Words and Phrases**
   A matcher is used that can convert a substring in the input to a canonical substring. The purpose of this step is to phrase the finding in words that NESTOR has indexed in its association (hash) files.

   ```
   Example:          HIGH CALCIUM
                       becomes
                  INCREASED CALCIUM
   ```

2. **Convert to a Property-Value Format**
   Each value that is not already paired with a property is used as a key to access a hashfile which contains every possible property that can be associated with it. The final result is a list of the form:

   $(((properties\text{-}of\text{-}value_1)\ value_1)$
   $\vdots$
   $((properties\text{-}of\text{-}value_n)\ value_n))$

   ```
   Example:

      INCREASED CALCIUM
           becomes
      (((WEIGHT LEVEL RATE) INCREASED) ((OBJECT) CALCIUM))
   ```

3. In this example we see that CALCIUM is always considered an OBJECT, however, INCREASED can be the value of the property WEIGHT, LEVEL, or RATE. We intend INCREASED to be the value of the LEVEL property. In the next step we will see how NESTOR finds this interpretation.

4. **Search for the Best Match**
   At this stage a matching function is called with every possible property-value pairing. In the example this would lead to the following 3 function calls:

```
MATCH( ((WEIGHT INCREASED)(OBJECT CALCIUM)))
MATCH( ((LEVEL INCREASED) (OBJECT CALCIUM)))
MATCH( ((RATE INCREASED)  (OBJECT CALCIUM)))
```

The MATCH function first sorts the property-value pairs in the argument list according to a predefined, static ranking of the properties. This is a purely heuristic ranking which is intended to place the more basic properties of the finding first; for example the object of a finding (e.g., calcium) would appear in the ranking before its location (e.g., the blood). A partial list of this ranking follows:

OBJECT PROCESS LOCATION STRUCTURE SUBSTANCE DISTRIBUTION DENSITY SOURCE CONTAINS QUALITY QUANTITY AGE CHARACTER COLOR DEGREE DIFFICULTY STRENGTH VOLUME WEIGHT

In the first MATCH call the list would be sorted to yield ((OBJECT CALCIUM)(WEIGHT INCREASED)) since the property OBJECT is considered more basic than the property WEIGHT.

With this sorted list the MATCH function begins processing the property-value pairs from the left to right. It accesses from a hashfile the findings associated with property-value$_i$ and intersects them with the findings matching all of the i-1 previous property-value pairs. If this intersection is nil then it stops and i-1 becomes the score of this interpretation of the finding. If this score is greater than the best score so far, then this interpretation becomes the new best one. The sorted lists scored by MATCH and their corresponding scores are as follows:

```
((OBJECT CALCIUM)(WEIGHT INCREASED))    Score: 1
((OBJECT CALCIUM)(LEVEL INCREASED))     Score: 2 <== winner
((OBJECT CALCIUM)(RATE INCREASED))      Score: 1
```

In the example, the second interpretation wins, which is what we had originally intended. Although the search procedure is exhaustive, it rarely takes more than one or two seconds since a finding will usually consists of no more than 4 values, and each value will usually have no more than 4 possible properties.

The winning interpretation may have only one finding associated with it, in which case the user is asked to verify that this is the one intended. However, it may be that there is more than one finding remaining, as in the winning interpretation above. This leads to the next step.

5. **Discriminate Among the Remaining Findings**

At this stage NESTOR has a set of findings, all of which match the criteria given by the user. In order to arrive at the one finding the user intends it is now necessary to ask the user some questions that will distinguish among the remaining findings. To do this NESTOR uses a predefined list of ranked properties to determine the property of highest importance which distinguishes among any of the findings. It then asks the user to select the value of this property from among a multiple choice list. This process is repeated until either there is only one finding, or there are no more properties that can distinguish among the remaining findings. If there is one finding remaining the user is asked to verify it. If there is more than one then the user is asked to select the ones desired.

```
Example:

The specification ((OBJECT CALCIUM)(LEVEL INCREASED))
has 2 findings associated with it:

((OBJECT CALCIUM)(LEVEL INCREASED)(LOCATION BLOOD))
((OBJECT CALCIUM)(LEVEL INCREASED)(LOCATION URINE))


In this case it is apparent that the only distinguishing
property among these two findings is LOCATION. So, NESTOR
asks the user to specify whether the LOCATION is in the
BLOOD or the URINE. Once the user specifies the location,
this leaves only one finding, and therefore the process
of entering this finding is complete.
```

At this point it is appropriate to question whether there are any significant differences between NESTOR's finding entry procedure and that of a keyword matching approach. Although both of them share the process of successively intersecting findings, there is one important difference. NESTOR pairs keywords, that are finding values, with their associated properties (the properties may be implied or keywords also). The pairing of properties with values is important in two stages in the algorithm. First, it influences the score of a given interpretation of the user's input by rank ordering the property-value pairs before scoring them. This has the effect of selecting those findings that are most likely the ones the user intended. Second, if more than one finding remains after all the user information has been utilized, NESTOR uses the rank ordering of properties to select the most important property to ask about in order to narrow the finding set further. This has the effect of making NESTOR appear more logical in its request for information. For example it would ask for the *object* of a finding before requesting its *location*.

3.1.3. Knowledge Aquisition of Nodes

In NESTOR the nodes are the first knowledge structures that must be acquired. The entry is straightforward. A prompt is made for each of the possible slots of a node such as its name, properties, units, etc. A disease requires some special information such as a prior probability, and etiology nodes. The slot containing the findings of a disease node is later created by NESTOR when the knowledge-base is indexed. Once nodes are defined, they may be used to define links as outlined in the next section.

3.2. Links

Currently, all of the links in NESTOR are causal links, and it is designed to be most useful in areas of medicine where the important causal links are known. For NESTOR's purposes an *important* link is one which if absent would lead to invalid scoring of some hypotheses given some set of findings. However, NESTOR does not require that the detailed mechanism *between* all nodes be known.

The remainder of this section will explain the representation and acquisition of causal links in NESTOR.

3.2.1. The Representation of Links

3.2.1.1. A Two Level Hierarchical Model

There are two different types of links in NESTOR, corresponding to the two level hierarchy of its causal model. Patil has previously used a multi-level causal representation in the ABEL program which performs medical diagnosis [Patil 81]. ABEL uses many causal levels and emphasizes qualitative, categorical reasoning. NESTOR emphasizes the use of probabilities, as well as categorical knowledge, in causal reasoning. NESTOR uses only two causal levels because this is the minimum needed to explore the combination of quantitative knowledge (at the top level) with qualitative knowledge (at the bottom level); the inclusion of additional causal levels is an area for future research.

As an example of NESTOR's two-level model, Figure 3-5 shows how serum calcium

causally relates to some other nodes in NESTOR's knowledge base. The upper level of the model contains probabilistic links between cause and effect nodes. The P(*) label on these links indicates that there is a probability density function (PDF) relating the cause to the effect. The nodes in the upper level are either etiologies or findings. The bottom level of the causal model contains, in addition, nodes which are intermediate causal states or processes and are not clinically observable. These are called intermediate nodes because they are intermediate causal ties between findings in the upper level. The functional form of the causal relationships at this level is indicated by a + (-) for those in which the effect variable is a monotonically increasing (decreasing) function of the cause variable. The dashed lines in the Figure 3-5 connect equivalent nodes between the two levels and exist only to help the reader visualize the correspondence between the two levels.

Notice in Figure 3-5 that the nodes are not specified in terms of specific values such as *increased serum calcium causing muscle weakness*, but rather in terms of a probabilistic relationship between the variables *serum calcium level* and *muscle strength*. This is largely because, in general, cause and effect nodes are multivalued rather than simply binary.

3.2.1.2. Link Restrictions in NESTOR

In the ideal situation NESTOR would have a multivariate PDF which for a local JCP related every set of causes to its set of effects. Unfortunately, this detailed knowledge is rarely known. This reality has greatly influenced the architecture of the causal links in NESTOR.

In the vast majority of areas of medicine if a conditional probability is known at all, then it is the conditional probability that relates a single cause to a single effect.[26] Consequently, NESTOR currently provides a language only for links of this type. Section 4.6.3 discusses some issues related to extending NESTOR to handle multivariate causation.

---

[26] The source of these probabilities are commonly either the medical literature or estimates from expert clinicians

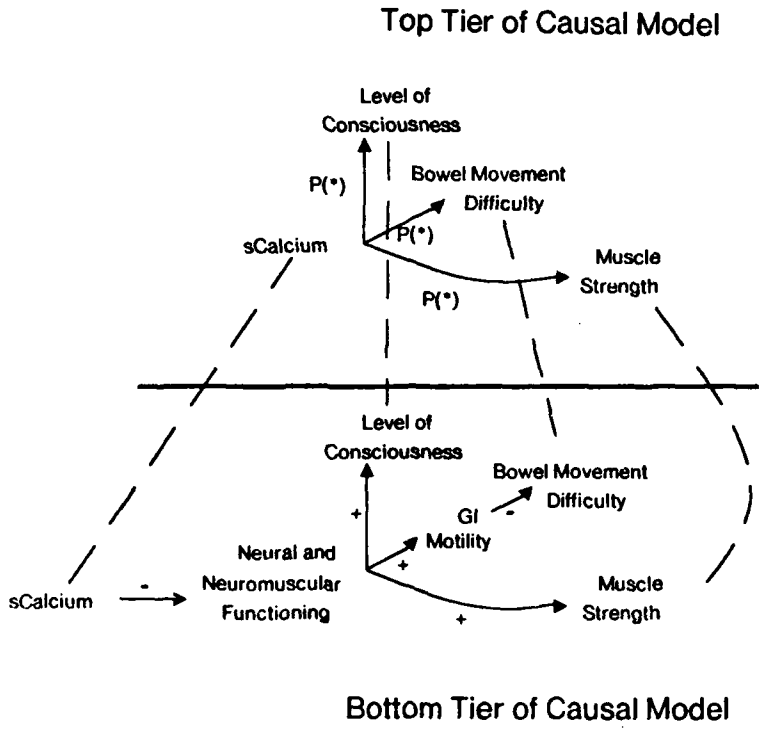


Figure 3-5: A Two Level Causal Model of Some of the Effects of Hypercalcemia

In order to handle those cases of multivariate causation which *do* occur, NESTOR has a means of combining the PDF's of several single links into a single multivariate PDF. This procedure will be covered in detail in Chapter 4. For now, the important point is that this combining procedure requires that the causal links be classified according to whether the effect variable is a monotonically increasing or decreasing function of the cause variable, given that other causal influences on the effect are held constant. For example, the serum calcium level is a monotonically increasing function of the serum PTH level. Obviously, this functional classification relies on the assumption that a cause always has one of these two relationships to its effect. Even so, this assumption appears to be valid for all the causal links within hypercalcemia that I have investigated. Perhaps a more complex class of

functions will be necessary in other domains of application, particularly those domains in which detailed mathematical relationships are known between the variables.

Another simplification of the link structure in NESTOR involves its assumptions about time. NESTOR currently is restricted to representing relationships in which the value of an effect occurs at the present moment, t = now, and this effect is a function only of the value of its cause at time t - ϵ, where ϵ is an infinitesimally small constant. A pragmatic reason for this restriction is that almost no temporally qualified probabilities are available in the literature, and estimation of them by my collaborating expert has been difficult. An expert has problems expressing this knowledge for at least two reasons. First, temporal patterns are complex and therefore storage and recall from human memory are difficult. Second, often even experts do not have exposure to the data needed to form temporally qualified probabilities. This is because for many causal variables the expert can not know the values of those variables which existed *before* the patient came for help. This makes it very difficult to acquire the probabilities for causal processes in which there is a long delay between the cause and the effect. A more theoretic reason for simplifying the temporal representation in NESTOR is that allowing every temporal causal pattern to be legal would greatly increase the size of the simulation space that NESTOR would need to explore in scoring an hypothesis. This point will be expanded in Chapter 4, but to get a basic understanding of it, consider two binary variables that are causally related. Without time there are only four possible states of the two binary variables. However, allowing the two variables to be individually qualified by their time of occurrence creates an infinite number of possible states.

The temporal simplifications made in NESTOR do restrict some of the links that can be represented in hypercalcemia. For example, currently the relationship between increased serum calcium and renal stones can not be represented since the former causes the latter only after some delay in time. However, fortunately many of the links in hypercalcemia can be represented within the current framework.

There is a great need for more work in the area of temporal representation in medicine. In particular more research is needed in order to find a restricted class of temporal patterns that is adequate for representing the *clinically important* temporal relationships among

causal variables. Such a restricted class might narrow the causal simulation space enough to make scoring of even complex, time-qualified diagnostic hypotheses computationally feasible. Section 4.6.6 contains additional discussion of some issues related to future research in the representation of temporal knowledge.

3.2.1.3. A Link Viewed as a Probability Density Function

The restrictions outlined in the last section enable NESTOR to represent the probabilistic relationship between a cause and effect node as a PDF in a tabular format. Table 3-1 is a PDF that relates the serum calcium level[27] to the level of consciousness which it affects.[28] Each element of the table has two numbers associated with it. The top number is the upper bound of the probability of the effect given the cause. The bottom number is the lower bound. These numbers were obtained from a specialist in endocrinology. Note that the table relates a discretized continuous causal variable (serum calcium level) to a discrete effect variable (level of consciousness). The PDF's in NESTOR are always represented as the relationship between two discrete variables. The next section discusses how this table was created.

3.2.2. The Acquisition of Links

The source of the probabilistic link information in NESTOR can be either subjective estimates or objective data. The links in the hypercalcemia knowledge base were obtained largely from expert estimates. For a particular value of a cause the expert specifies the upper and lower bound on the probability of each possible effect value. The capability to express upper and lower probability bounds allows the expert to convey an appropriate level of

---

[27] This assumes that the total serum calcium level has been adjusted on the basis of the serum albumin level.

[28] The need for some modifications to the entries in this table was apparent after the experiment discussed in Chapter 4 indicated that the expert was not in full agreement with some of them. Also, the entries for a serum calcium of 0 to 9 require modification if NESTOR is used to diagnose hypocalcemic disorders. However, the table is basically correct, and is shown here in unmodified form since it was used in the experiment that tested NESTOR's performance in scoring hypotheses. Furthermore, the principles underlying the representation and acquisition of the table are the same regardless of the specific table entries, and this is the main point for now.

Alert	100	100	100	99	80	50	0	0	0	0	0	0
	100	100	99	80	50	0	0	0	0	0	0	0
Lethargy	0	0	1	20	50	90	90	60	30	0	0	0
	0	0	0	1	26	50	53	27	0	0	0	0
Stupor	0	0	0	0	0	20	33	47	60	60	20	0
	0	0	0	0	0	0	10	20	30	20	0	0
Coma	0	0	0	0	0	0	20	40	60	80	100	100
	0	0	0	0	0	0	0	13	27	40	70	100
	0-9	9-10.5	10.5-12	12-13	13-14	14-15	15-16	16-17	17-18	18-19	19-20	20-30

Serum Calcium Level (mg/100ml) -->

Table 3-1: The PDF Relating Serum Calcium Level to Level of Consciousness

uncertainty about the probabilities themselves. It releases the expert from the chore of having to decide on one exact number to represent a probability, and it increases the likelihood that the probabilistic knowledge is valid.

The following transcript shows the entry of part of the information that defines the link between the serum calcium level and the level of consciousness The user's input is underlined.

```
LINK TASK: DEFINE

IS THE LINK IN THE TOP OR BOTTOM LEVEL
   OF THE MODEL'S HIERARCHY? (T/B): T
IS THE RELATIONSHIP BETWEEN THE CAUSE AND EFFECT
   NODE MONOTONICALLY INCREASING, DECREASING,
   OR UNKNOWN? (I/D/U): D

CAUSE VARIABLE:
** SERUM CALCIUM

THE BEST MATCHES TO YOUR DESCRIPTION FOLLOW:
1) TOTAL SERUM CALCIUM
IS THIS OK? (Y/N): Y

EFFECT VARIABLE:
** CONSCIOUSNESS

THE BEST MATCHES TO YOUR DESCRIPTION FOLLOW:
1) LEVEL OF CONSCIOUSNESS
IS THIS OK? (Y/N): Y

     LOW VALUE OF TOTAL SERUM CALCIUM  (IN MG/100ML): 16
     HIGH VALUE OF TOTAL SERUM CALCIUM  (IN MG/100ML): 16

        LEVEL OF CONSCIOUSNESS: COMA
          LOW PROBABILITY: 0
          HIGH PROBABILITY: 0

        LEVEL OF CONSCIOUSNESS: STUPOR
          LOW PROBABILITY: 10
          HIGH PROBABILITY: 20

        LEVEL OF CONSCIOUSNESS: LETHARGY
          LOW PROBABILITY: 80
          HIGH PROBABILITY: 90

        LEVEL OF CONSCIOUSNESS: ALERT
          LOW PROBABILITY: 0
          HIGH PROBABILITY: 0
```

NESTOR performs a consistency check on the probabilities by calculating whether the sum of the lower bounds for a given column of the PDF table (such as serum calcium = 12 to 13) is $\leq 100\%$ and the sum of the upper bounds is $\geq 100\%$[29]. If there is an error the expert is told about this and then allowed to edit the probabilities to achieve consistency.

---

[29] In some cases, such as with binary variables, other consistency checks could be performed, but currently are not.

NESTOR does not require exhaustive entry of the PDF. Experts are allowed to enter link probabilities for just those values of the causal node they consider important. The only requirement is that the endpoints must be entered (in the example these are the columns associated with a serum calcium of 0 and 30). Linear interpolation between known values of *bounds* is used to determine values not entered. It is important to realize that the interpolation is being done from one lower bound to another, and from one upper bound to another; if the expert does not agree with the resulting interpolated bounds then they can be edited. Table 3-2 shows the values that the expert entered for the example. The values for a serum calcium level of 1 to 10, 16, 17, and 19 have not been entered by the expert; the absence of the latter three entries are explicitly shown in the table as empty columns.

Alert	100	100	99	80	50	0			0		0	0
	100	100	99	80	50	0			0		0	0
Lethargy	0	0	1	20	50	90			0		0	0
	0	0	1	20	50	80			0		0	0
Stupor	0	0	0	0	0	20			60		0	0
	0	0	0	0	0	10			40		0	0
Coma	0	0	0	0	0	0			60		100	100
	0	0	0	0	0	0			40		100	100
	0	11	12	13	14	15	16	17	18	19	20	30

Serum Calcium Level (mg/100ml) -->

Table 3-2: Partial PDF Relating Serum Calcium Level to Level of Consciousness

Table 3-3 shows the results after the interpolation of missing values.

Once this table is computed, NESTOR creates a table in which both the cause and effect variables are discretized into intervals that are defined by slots in their respective nodes. Table 3-1, shown previously, was created by this process.

The bounds in Table 3-1 emanate from two sources. First, the user originally introduced some bounds. Second, the process of discretizing the causal variable results in having to set the bounds of any interval to the maximum and minimum values within that interval. In order to tighten some of the bounds in the table, tighter intervals could be used However, the above intervals are sufficient for current purposes.

Alert	100	100	99	80	50	0	0	0	0	0	0	0
	100	100	99	80	50	0	0	0	0	0	0	0
Lethargy	0	0	1	20	50	90	60	30	0	0	0	0
	0	0	1	20	50	80	53	27	0	0	0	0
Stupor	0	0	0	0	0	20	33	47	60	30	0	0
	0	0	0	0	0	10	20	30	40	20	0	0
Coma	0	0	0	0	0	0	20	40	60	80	100	100
	0	0	0	0	0	0	13	27	40	70	100	100
	0	11	12	13	14	15	16	17	18	19	20	30

Serum Calcium Level (mg/100ml) -->

Table 3-3: Complete PDF Relating Serum Calcium Level to Level of Consciousness

Table 3-1 can now be efficiently used to determine the probability of a particular level of consciousness given a particular level of serum calcium.

3.2.3. The Use of Links

The PDF tables express the probabilistic relationships between nodes in a causal model. This section discusses how these tables are used to compute a conditional probability of an effect given a cause.

3.2.3.1. Computing Conditional Probabilities

Computing a conditional probability of an effect value given a cause value is an important calculation in NESTOR. In general the value of a cause node can be an interval rather than an exact value. Thus, a serum calcium of 12.5 to 13 mg/100ml is a legal value for the serum calcium node. Suppose the probability of lethargy given a serum calcium of 12.5 to 13 mg/100ml is needed. In this case the value 12.5 to 13 falls within an interval 12 to 13 in Table 3-1, and therefore the probability by table lookup is 1% to 20%.

In another case suppose the serum calcium is 13.5 to 14.5 mg/100ml. In this situation the value 13.5 to 14.5 overlaps the intervals 13 to 14 and 14 to 15. So, to compute the

probability of lethargy given a calcium value of 13.5 to 14.5, the lower bound is the minimum lower bound of the two intervals (26%) and the upper bound is the maximum upper bound of the two intervals (90%).

The value of the effect node can also be an interval (or a disjunctive set of values if the effect node has discrete values). Thus, the probability of coma *or* stupor given a serum calcium value of 17 to 19 mg/100ml is 47% to 100%. The lower bound was found by summing the lower bounds of stupor (20%) and coma (27%). Similarly, the upper bound was derived by summing the upper bounds of stupor (60%) and coma (80%), then applying the fact that no probability can be greater than 100%.

3.2.3.2. The Importance of Probabilistic Bounding in Representing Conditional Probabilities

The ability to represent a probability with an upper and lower bound is an important feature of NESTOR. The Dempster-Shafer theory of evidence [Shafer 76] also uses bounds in describing the likelihood of a proposition (e.g., a diagnosis), and the theory has been applied in a number of computer systems [Barnett 81, Garvey 81]. With regard to the issue of bounding, the Dempster-Shafer theory emphasizes the bounding of the probability of derived propositions. NESTOR, however, also allows the probabilities of base propositions to be bounded. This means that experts entering the conditional and prior probabilities for a given domain can bound them rather than specify them exactly. This increases the likelihood that the knowledge is a valid reflection of the population statistics that are being estimated by the experts.

In addition to providing flexible knowledge-acquisition of probabilities, the ability to bound the probabilistic knowledge has other advantages. We have seen previously how the value of a cause or effect node can be an interval of values. The ability to validly represent a conditional probability P(A | B), where A or B is a discretized continuous variable, depends on bounding P(A | B). To see why this is true, suppose all we know is that the value of the serum calcium is between 13 and 14 mg/100ml. Furthermore, suppose we *do not* know the probability distribution of the values between 13 and 14 for this population. In this case it is not possible to specify the exact probability of lethargy given a serum calcium of 13 to 14,

because the probability of lethargy given a calcium of 13 may be different from the probability if it is 14 (or anywhere in between). However, the probability of lethargy given a calcium between 13 and 14 *can* be bounded by determining the minimum probability of lethargy for *any* calcium value between 13 and 14, and similarly determining the maximum probability. Thus, it is not necessary to assume any particular probability distribution for the calcium values between 13 and 14.

To understand further the interlinkage of probability bounding and a lack of information about the probability distribution, consider the following common situation. A probability-based reasoning system has a set of discretized continuous variables (often simply binary variables). The values of a continuous variable, such as the serum calcium level, might be discretized with the values of LOW, NORMAL, and HIGH. Suppose the following probabilities are available:[30]

```
P(COMA | Ca = HIGH) = 50%
P(Ca = HIGH | PHPT) = 95%
P(Ca = HIGH | MYELOMA) = 30%
```

Here PHPT and MYELOMA are two diseases that cause hypercalcemia. Suppose a patient is in a coma. In order to determine, based on this data, which of these two diseases is most likely, we need to first determine the probability of coma given a disease. Two straightforward calculations yield the following results:

```
P(COMA | PHPT) = P(COMA | Ca = HIGH) x P(Ca = HIGH | PHPT)
               = 0.5 x 0.95 = 47.5%

P(COMA | MYELOMA) = P(COMA | Ca = HIGH) x P(Ca = HIGH | MYELOMA)
                  = 0.5 x 0.3 = 15%
```

Thus, from these results we conclude that coma is more common in PHPT than in MYELOMA. But, this is wrong. The catch is that each of the three conditional probabilities assumes a different distribution of high calcium values. For P(COMA | Ca = HIGH) the distribution of high calcium values might have been taken from the entire population of patients with a high serum calcium. For P(Ca = HIGH | PHPT) and P(Ca = HIGH | MYELOMA) the distribution of high calcium values is that of each disease respectively. However, these two distributions are not the same. Myeloma causes higher

---

[30]The precise value of these conditional probabilities are used for illustration purposes only and were not acquired from either a database or an expert.

serum calcium levels than PHPT, although PHPT more often causes *some* degree of abnormal increase in the calcium level. It is the higher serum calcium levels that lead to coma. Thus, coma is actually more likely to be seen in myeloma than in PHPT. The reason the above calculations lead to the wrong conclusion is that the implied distribution of Ca = HIGH was different in each of the three conditional probabilities.

This dilemma is circumvented in NESTOR by having bounds on the conditional probabilities. The lower bound represents the lowest probability of the effect (in this case coma) given *any* value of the cause (high calcium) within the given interval. The upper bound is similarly defined. This representation method makes it unnecessary to know the probability distribution of the cause variable, since the probability of the effect has been bounded by considering *every* possible distribution of the causal variable. If the resulting bounds are considered too wide for practical use, then the intervals of the variables can be narrowed; for example, the single large interval ***high calcium*** (i.e., serum calcium $\geq$ 10.5) could be divided into a number of smaller intervals, each with a range of 1 mg/100ml.

Chapter 4 discusses additional reasons why bounding probabilities is important in scoring a diagnostic hypothesis.

Chapter 4

Scoring a Diagnostic Hypothesis

One of NESTOR's primary tasks is to find the most likely causes of a given set of findings. To do this it is sufficient to rank order the diagnostic possibilities according to their posterior probabilities. This chapter explains how NESTOR computes a rank ordering.

To rank order the hypotheses a score is assigned to each hypothesis. The score is a number reflecting the likelihood of an hypothesis in the context of a given set of findings. Sorting the scores of a set of diagnostic hypotheses allows them to be rank ordered. This chapter will first discuss the nature of the score itself. Then the procedure used to derive the score will be explained, as well as related research. Finally, a formal evaluation of NESTOR's scoring method will be presented and extensions to it will be discussed.

4.1. The Scoring Metric

A score is simply a number assigned to a diagnostic hypothesis. The *meaning* of that number depends on the ***scoring metric*** being used. For example, a posterior probability is a scoring metric that expresses the probability of a diagnostic hypothesis given a set of findings. Many current computer diagnostic aids use an ad hoc scoring metric. For example, INTERNIST uses a linear sum of evoking, frequency, and importance weights as the primary basis of its scoring metric [Miller 82].

In NESTOR there are three major requirements for a scoring metric:

1. The rank order of a set of hypotheses that is determined on basis of the scoring metric must be equivalent to the rank order of the hypotheses that is determined on the basis of their posterior probabilities. Since bounded probabilities are used in NESTOR, it is actually more precise to state that the *partial order* of the hypotheses are the same with both methods

2. It must be computable from sensitivities (the probability of a set of findings given a disease), and from the prior probabilities of the diseases. This is necessary since most probabilistic data in the literature is in this form, and additionally, experts are most comfortable estimating this type of probability.

3. Computing it must be computationally tractable with regard to computing time and space requirements.

4.1.1. Bayes' Formula

An expanded form of Bayes' formula, shown in Figure 4-1, suggests that the posterior probability on the left be used as a scoring metric by performing the calculation on the right hand side. In this formula F is a set of case specific findings, H_i is some diagnostic hypothesis being scored, and N is the total number of competing diagnostic hypotheses (including multiple disease hypotheses).

$$P(H_1 \mid F) = \frac{P(F \mid H_1) \times P(H_1)}{\sum_{j=1}^{N} P(F \mid H_j) \times P(H_j)}$$

Figure 4-1: An Expanded Form of Bayes' Formula

The posterior probability appears on the left side of the equation and is certainly adequate for rank ordering a set of hypotheses. So, the first requirement for a scoring metric is satisfied. The second requirement is also achieved since the right hand side of the equation is expressed solely in terms of sensitivities and prior probabilities. However, one major problem with using Bayes' Formula lies in the denominator. The denominator introduces the following two conditions:

1. ***Exhaustiveness of Diseases***
   The sum in Figure 4-1 must occur over all possible hypotheses H_i in order for the denominator to be valid. H_j includes not just all potential single disease

diagnoses, but multiple ones as well. The number of hypotheses that must be scored and then summed is an exponential function of the number of diseases in the domain. For just 40 diseases this would require that $2^{40} = 10^{12}$ scores be computed. Clearly, this is not practically possible and therefore requirement 3 above is not satisfied.

To avoid this computational problem many systems assume that there is only one disease present. However, this is often an invalid assumption, so the problem of computational tractability is solved at the expense of inaccurate scoring.

2. ***Mutual Exclusivity of Diseases***
   The summation in Figure 4-1 also assumes that the diagnostic hypotheses are mutually exclusive.

In order to see how to avoid the summation in the denominator of Figure 4-1 it is helpful to recall how that formula was derived. Figure 4-2 shows the equation from which the formula is derived. The formula follows simply from the definition of conditional probabilities. Notice that the only difference between Figure 4-1 and Figure 4-2 is the denominator. In essence the denominator in Figure 4-1 provides a means of calculating P(F) in terms of sensitivities ($P(F|H_i)$) and priors ($P(H_i)$). Note that P(F) is a *constant* for any given set of findings F. This fact is the key to the scoring metric used by NESTOR.

$$P(H_1 \mid F) = \frac{P(F \mid H_1) \times P(H_1)}{P(F)} = \frac{P(F \,\&\, H)}{P(F)}$$

Figure 4-2: Definition of Posterior Probability

4.1.2. NESTOR's Scoring Metric

Since P(F) is constant during the rank ordering of a set of diagnostic hypotheses, it may be removed to yield $P(F \mid H_i) \times P(H_i)$ as the scoring metric. Figure 4-3 shows why this is so.

R1. $P(F \mid H_a) \times P(H_a) \geq P(F \mid H_b) \times P(H_b)$

implies that

R2. $$\frac{P(F \mid H_a) \times P(H_a)}{P(F)} \geq \frac{P(F \mid H_b) \times P(H_b)}{P(F)}$$

implies that

R3. $P(H_a \mid F) \geq P(H_b \mid F)$

Figure 4-3: The Adequacy of $P(F \mid H_i) \times P(H_i)$ for Rank Ordering

R2 results from dividing R1 by the constant P(F). Since P(F) is positive this division does not alter the direction of the inequality. R3 results from applying the definition of posterior probabilities (see Figure 4-2) to the two terms in R2. Thus, R1 implies R3, which means that $P(F \mid H_i) \times P(H_i)$ can serve as a scoring metric to rank order diagnostic hypotheses by their posterior probabilities. This is the scoring metric used by NESTOR, and it will be called SM.

Note that SM satisfies the original three criteria for a scoring metric, namely:

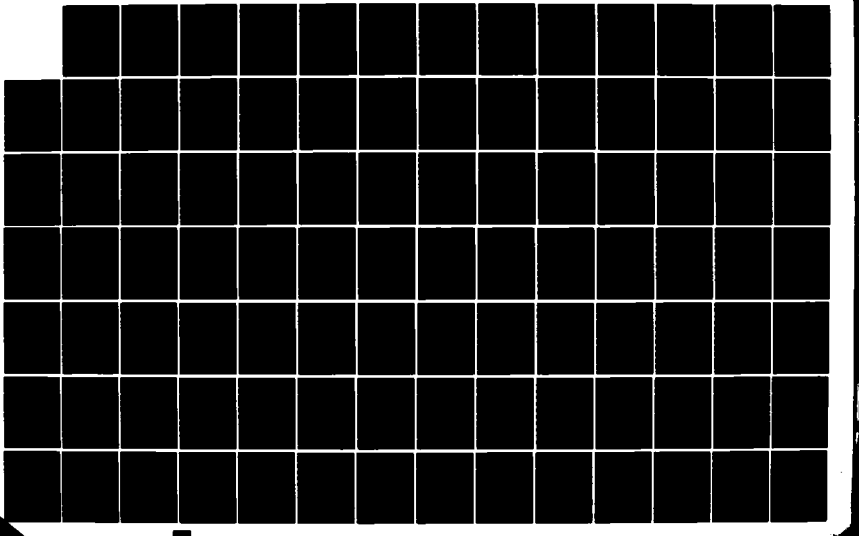
1. It can be used to rank order diagnostic hypotheses by their posterior probabilities.

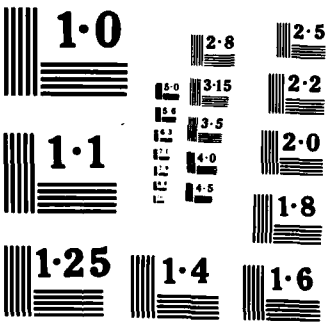
2. It uses only sensitivities and prior probabilities.

3. Computing it is computationally tractable. This is because for a given set of

AD-A152 046 NESTOR: A COMPUTER-BASED MEDICAL DIAGNOSTIC AID THAT INTEGRATES CAUSAL AN..(U) STANFORD UNIV CA DEPT OF COMPUTER SCIENCE G F COOPER NOV 84 STAN-CS-84-1031 2/3

UNCLASSIFIED N00014-81-K-0004 F/G 9/2 NL




1·0
1·1
1·25
2·8
3·15
3·5
4·0
4·5
1·4
2·5
2·2
2·0
1·8
1·6

hypotheses of size N, only N scores have to be computed, rather than scoring every possible hypothesis. This is obviously useful when physicians have only a small number of diagnoses in their differential and they would like NESTOR to rank order them. In addition, Chapter 5 discusses how NESTOR takes advantage of the properties of SM to efficiently find the top few hypotheses from among all possible hypotheses.

Also, SM avoids the conditions of *exhaustiveness* and *exclusivity* that occur in applying the formula in Figure 4-1, since the denominator of that formula, which was responsible for their introduction, does not appear in SM.

The disadvantage of using SM is that it does not convey the probability of the most probable hypothesis. This is especially undesirable when the most probable hypothesis is not very probable. The current chapter does not address this problem, but Section 6.2 of Chapter 6 will discuss a method for iteratively tightening the bounds on the *posterior* probability of the most probable hypothesis by successive calculations of the SM scores of other hypotheses.

4.1.2.1. The Causal Interpretation of the Scoring Metric

Since $P(F \mid H_i)$ is a term in SM, its interpretation is central to the interpretation of SM. A *causal* interpretation is placed on this conditional probability in NESTOR. NESTOR assumes that the diseases in a diagnostic hypothesis H are meant to causally justify the findings in F. That is, NESTOR always assumes that a *causal* hypothesis is being scored. To see the implications of this, consider the following example shown in Figure 4-4.

Suppose the etiology of disease D_a is the only cause of the etiology of disease D_b. Furthermore, D_a also causes the findings in F. D_b is not causally related to the findings in F. Now, with reasonable assumptions $P(F \mid D_b) > 0$ since D_b suggests the existence of D_a and D_a suggests the existence of F. However, strictly speaking, NESTOR does not calculate $P(F \mid D_b)$. Instead, it calculates $P(D_b$ is causing $F \mid D_b)$, which in this example is equal to zero because D_b can not *cause* F. Therefore, whenever $P(F \mid D_i)$ is used it should be taken as an abbreviation for $P(D_i$ is causing $F \mid D_i)$. Similarly, since $P(F \ \& \ D_i) =$

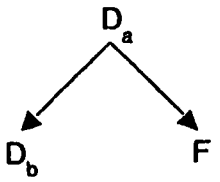


Figure 4-4: A Causal Graph

$P(F \mid D_i) \times P(D_i) = SM$, the term $P(F \& D_i)$ should be taken to mean $P(D_i$ is causing F). These causal interpretations are usually intended in CAMDM programs, but they are rarely stated explicitly.

4.1.2.2. Bounding the Scoring Metric

Most CAMDM programs derive a *single* number which expresses the score of an hypothesis. The number may be a probability or it may be ad hoc. In NESTOR the goal of scoring an hypothesis is relaxed to that of merely placing an upper and lower bound on the score of the hypothesis. If r1, r2, and r3 are real numbers between 0 and 1, then NESTOR concludes that $r1 \leq P(F \& D_i) \leq r2$, instead of attempting to be exactly precise by concluding $P(F \& D_i) = r3$. There are two main advantages to bounding SM.

1. The precision (or lack thereof) of NESTOR's knowledge of conditional and prior probabilities is appropriately reflected in the precision of SM. Thus, the precision of SM is a function of the precision of the knowledge that supports it. This allows precise results (in the form of tight bounds on SM) when there is precise knowledge, yet it does not sacrifice valid results[31] in the face of imprecise knowledge. Instead, the precision of the results decreases just enough to maintain accuracy.

[31] See page 17 for a definition of *validity* of a probability in NESTOR.

2. Causal knowledge can sometimes be combined with probabilistic knowledge to accurately *bound* SM in light of available probabilistic knowledge, but not to calculate SM exactly. This point is developed in detail later in this chapter.

Therefore, the reason NESTOR bounds SM is because in general it is much easier to maintain the validity of a bounded scoring metric, than a point scoring metric.

4.2. The Scoring Algorithm: An Overview

The goal of the scoring algorithm is to receive a diagnostic hypothesis and return the bounds on its score. To do this, NESTOR uses the scoring metric $P(F \& D_i)$, called SM. The ability to score hypotheses means that for a given patient case NESTOR can take a large set of possible diagnostic hypotheses, rank order them by their respective scores, and return the most likely hypotheses (i.e., the ones with the highest scores). This section will overview how NESTOR uses causal and probabilistic knowledge to calculate SM.

4.2.1. The Nature of Causal Knowledge in NESTOR

Causal knowledge in NESTOR is in the form of causal links that interconnect nodes which represent states or processes. Later in this chapter we will see how a causal graph is created from a set of findings. This graph connects the etiologies (of the hypothesis being scored) to the findings through possibly many intermediate causal nodes.

A fundamentally important aspect of a causal representation is that it greatly limits the consideration of the possible influences on any node's value. For example, Figure 4-5 shows a causal graph connecting the etiology of PHPT, namely increased PTH, to three of its effects. Note how this graph makes immediately clear the possible influences of the four nodes on each other. For example, it is apparent that the urinary calcium level can not affect the level of consciousness, nor vice versa. Nor can either of them affect the calcium level. This localization of influences allows NESTOR to apply its probabilistic knowledge in a focused manner.

Since every conditional probability in NESTOR also expresses a causal relationship, it is easy to confuse the two. However, the causal knowledge should be appreciated as being

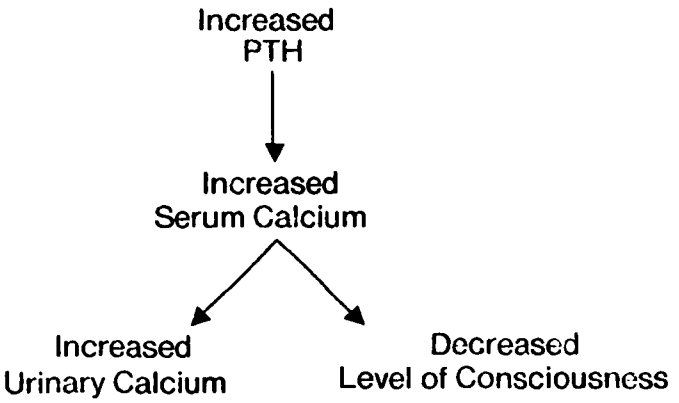


Figure 4-5: Causal Relationships Among Four Nodes in PHPT

extra-probabilistic. The key utility of causal knowledge in NESTOR is the way in which it guides the interpretation of probabilistic knowledge.

The ***context*** in which the causal links are presumed to exist is just as important as the links themselves to the validity of the scoring process. The context contains all the significant links that are *not* explicitly represented because they are assumed to have a constant influence on the nodes that *are* represented. Their influence on the nodes in any causal graph is factored into the links that are represented. For example, the fact that the patient has normal mental alertness, outside of any calcium influences, is assumed in Figure 4-5. Its existence has been factored into the probability relating the serum calcium level to the level of consciousness. If the patient had recently had a severe head injury, then this causal graph would be incomplete and possibly inaccurate; this problem is solved by explicitly representing head injury as a finding or an hypothesized etiology of the current set of findings. In fact every normal physiological process of the human body is a potential set of links which are not represented in NESTOR unless they are known to significantly influence the disease being scored. Therefore, every causal link in NESTOR assumes a context in which every other influence on the effect node is physiologically normal.

4.2.2. Causal simulation

The scoring procedure used by NESTOR can be viewed as a simulation of the state space of a disease-specific causal graph. In the graph the values of the finding and etiology nodes are held constant, while all the intermediate causal nodes connecting them take on every possible value within their defined range.[32] Each state in the simulation space corresponds to a unique set of values assigned to the intermediate causal nodes. The probability of each possible state (of values) is computed. When these individual state probabilities are summed, the result is the probability of observing the findings occurring given the hypothesis. This is of course P(F | H). The intuitive justification is this: the result of summing the probability of every possible unique way in which an event (the findings given the hypothesis) can occur is the probability of that event occurring. The following equation expresses the technique in mathematical terms:

$$P(F \mid H_i) = \sum_{n=1}^{2^N} P(F \,\&\, G(n) \mid E_i)$$

Here H_i is the hypothesized set of diseases being scored, F is the set of findings, E_i is the set of etiological nodes of the diseases in H_i, and N is the number of intermediate causal states. The values of the intermediate causal nodes are assumed to be binary in this equation for simplicity. Since there are N intermediate binary nodes, there are 2^N possible states, because the members of F and E_i have fixed values. The function G maps an integer into a binary number, which is interpreted as a binary value assignment to each of the N intermediate nodes.

4.2.3. A Brief Example

An example will show how an hypothesis is scored in a three step process. The case consists of two findings, f_1 and f_2, that constitute the set F. The goal of the scoring algorithm is to compute $P(F \,\&\, H_a)$ for some hypothesis H_a. Since $P(F \,\&\, H_a) = P(F \mid H_a) \times P(H_a)$ and we assume $P(H_a)$ is known from available statistics or estimates, the task is to compute

---

[32] Section 4.6.1 discusses an efficient method of doing this using caching techniques.

$P(F \mid H_a)$. For the purposes of the example, assume that H_a consists of one disease D_a which has one etiology E_a.

The first step in the scoring process is to construct a causal graph, as shown in Figure 4-6, which connects f_1 and f_2 to the etiological node E_a. The graph is generated by starting with each finding and chaining backward to the etiological nodes of the diseases of the hypothesis being scored. The resulting graph contains every possible causal linkage (known to NESTOR) from E_a to f_1 and f_2. I_1 and I_2 are intermediate causal nodes.

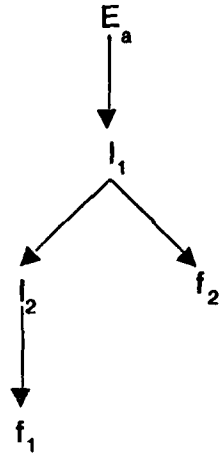


Figure 4-6: Step 1: Causal Graph Generated for the Example

The second step in scoring the sample hypothesis is to divide the causal graph into levels as shown in Figure 4-7.[33] The levels are created to insure that any node at level j is causally influenced only by nodes at a level less than j. This allows ordering of the scoring process in step 3.

[33]Note that the the term *levels* is being used differently here than it is in Chapter 3 where it refers to the two levels (tiers) of abstraction in NESTOR's causal model. The current use of the term *levels* refers to a grouping of nodes within a given plane of causal abstraction, whereas the use of *levels* in Chapter 3 refers to different planes of causal abstraction.

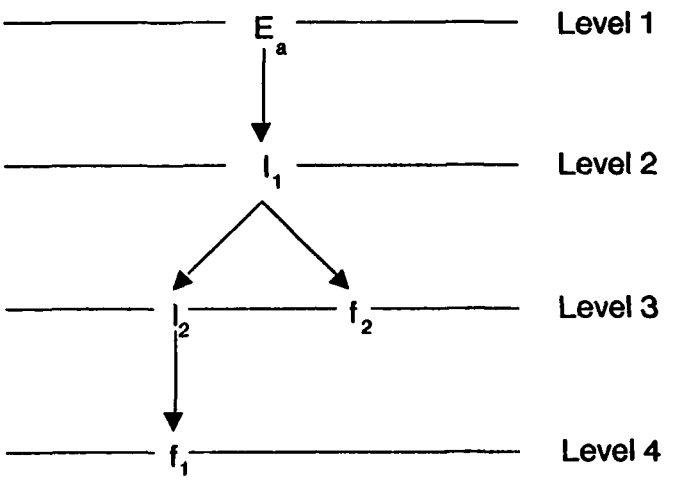


Figure 4-7: Step 2: Assigning Nodes to Levels in the Example

The third step in the scoring process is to iterate over all the possible values of the intermediate nodes computing the following:

$P(\text{value of nodes at level}_{j+1} \mid \text{value of nodes at level}_1 \text{ to level}_j)$

. This is the critical calculation of step 3. In what follows we will see how this readily leads to the calculation of SM for H_a. In this example, for clarity, all variables are assumed to be binary. The *value of a level* in the graph will be used to refer to the values of the nodes at that level.

Figure 4-8 shows the conditional probabilities associated with the links in the graph of the example. Precise values are used instead of bounded probabilities in order to make the calculations in the example easier to follow. A tilde before a node label means that the node has the value FALSE, otherwise its value is TRUE.

E_a ---> I_1:

$P(I_1 \mid E_a) = 0.8$ $\quad P(\sim I_1 \mid E_a) = 0.2$

$P(I_1 \mid \sim E_a) = 0.3$ $\quad P(\sim I_1 \mid \sim E_a) = 0.7$

I_1 ---> I_2:

$P(I_2 \mid I_1) = 0.4$ $\quad P(\sim I_2 \mid I_1) = 0.6$

$P(I_2 \mid \sim I_1) = 0.1$ $\quad P(\sim I_2 \mid \sim I_1) = 0.9$

I_1 ---> f_2:

$P(f_2 \mid I_1) = 0.6$ $\quad P(\sim f_2 \mid I_1) = 0.4$

$P(f_2 \mid \sim I_1) = 0.2$ $\quad P(\sim f_2 \mid \sim I_1) = 0.8$

I_2 ---> f_1:

$P(f_1 \mid I_2) = 0.7$ $\quad P(\sim f_1 \mid I_2) = 0.3$

$P(f_1 \mid \sim I_2) = 0.1$ $\quad P(\sim f_1 \mid \sim I_2) = 0.9$

Figure 4-8: Conditional Probabilities Used in the Example

Figure 4-9 shows one graph for each step in the iteration over the values of the levels. Each of these graphs contains a unique instantiation of node values. Note that the etiology node E_a and the finding nodes f_1 and f_2 have a constant value of TRUE. The relevant conditional probabilities are shown next to the links in the graphs. To score one of these graphs requires calculating the probability that the nodes in the graph will have the values they are assigned, given that the etiology exists. The sum of all the graph scores is 0.169, which is the value of $P(f_1 \,\&\, f_2 \mid E_a)$. The product of $P(f_1 \,\&\, f_2 \mid E_a)$ with $P(E_a)$ $(= P(H_a))$ is the score for H_a given f_1 and f_2.

The probability of a particular instantiation of values in a graph is called P_{in}. If there are N levels in a causal graph, then the P_{in} of the graph is calculated as follows:

$$P_{in} = \prod_{j=1}^{N-1} P(\text{node values at level } j+1 \mid \text{node values at levels 1 to } j)$$

Figure 4-9 shows the P_{in} value for each possible instantiation of the graph. In order to

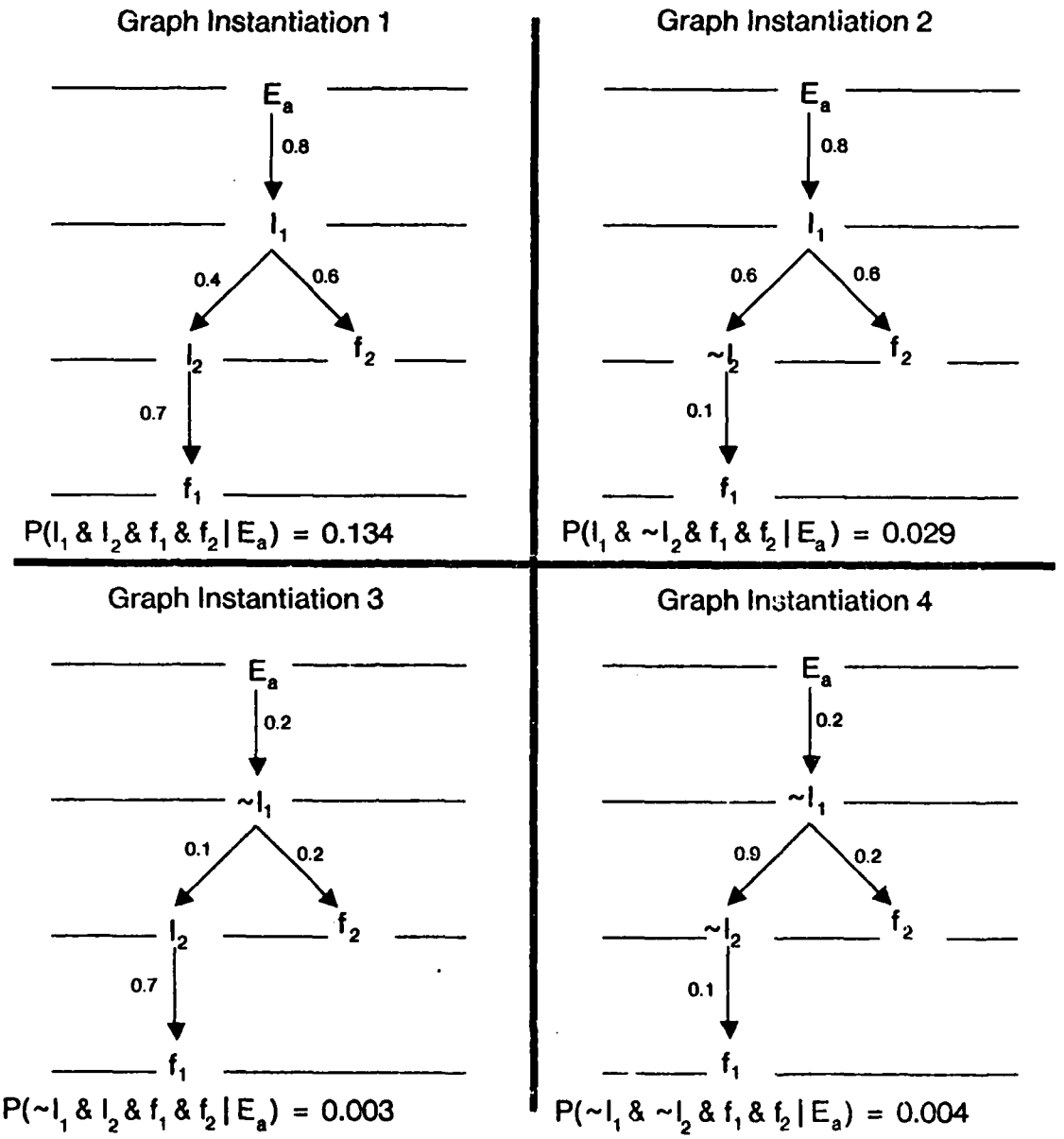


Figure 4-9: Step3: Calculating the Probability of Causal Graphs with Specific Value Instantiations for All Intermediate Variables

demonstrate the method of calculating the P_{in}'s, the P_{in} for the first graph will be derived. If the above equation is applied to graph 1 in Figure 4-9, the following equation results:

P_{in} = P(level 2 values | level 1 values)
x P(level 3 values | level 1 to level 2 values)
x P(level 4 values | level 1 to level 3 values)

Each of these three conditional probabilities is derived as follows:

P(level 2 values | level 1 values)

$= P(I_1 \mid E_a)$

$= 0.8$

Explanation: This is a given probability (see Figure 4-8).

P(level 3 values| level 1 to level 2 values)

$= P(I_2 \ \& \ f_2 \mid I_1)$

$= P(I_2 \mid I_1) \times P(f_2 \mid I_1)$

$= 0.4 \times 0.6$

$= 0.24$

Explanation: I_2 and f_2 are assumed to be independent given I_1. This is a form of default reasoning in which the effects of a cause are assumed independent unless some explicit causal connection between them is known. This is discussed in detail in Section 4.3.3.

P(level 4 values | level 1 to level 3 values)

$= P(f_1 \mid I_2)$

$= 0.7$

Explanation: This is a given probability (see Figure 4-8).

Thus, the P_{in} for graph instantiation 1 is equal to 0.8 x 0.24 x 0.7 = 0.134. The P_{in} values for the other three graphs are shown in Figure 4-9 and are 0.029, 0.003, and 0.004. The four instantiated graphs represent every possible way in which disease D_a can cause the findings f_1 and f_2. The sum of the four P_{in} values is 0.169, which is the value of $P(f_1 \ \& \ f_2 \mid D_a)$. If $P(D_a) = 0.1$, then the SM score of H_a is 0.169 x 0.1 = 0.0169, and this is the SM score for D_a causing f_1 and f_2. Figure 4-10 shows, within the context of medical

decision making, the steps just used in computing P(F & H), the score of hypothesis H given the findings F. In the next section these steps will be explained in detail.

4.3. The Scoring Algorithm: The Details

4.3.1. Step1: Create a Patient-Specific Causal Graph

The goal of this step is to create a causal graph that links the etiologies of the diseases of the hypothesis being scored to the patient findings. Recall that the findings are just nodes instantiated with specific values (e.g., serum calcium = 13 mg/100ml). The causal graph is constructed by chaining backward from the findings through all causal pathways until encountering the etiology nodes of the hypothesis being scored. If there is more than one disease in the hypothesis being scored, the etiology nodes are a union of all the disease etiologies. In this way, a case specific multi-disease causal graph can be constructed. Although etiologies usually occur at level 1, an etiology of one disease in an hypothesis may be causally affected by the etiology of another disease in the hypothesis and therefore would be at a level greater than level 1.

Normal physiology is always an etiological node in every hypothesis. If a finding can not be causally connected to one of the other etiologies of the hypothesis being scored, then it is connected to the *normal physiology* node. In this way those findings that are not associated with the diseases of the hypothesis begin scored and that can possibly be explained by normal physiology do not cause the hypothesis to receive a score of zero.

At this stage it is also possible to constrain the possible values of the nodes in the graph so that fewer values must be instantiated in step 3, and thus less computation performed. This is done by having the possible values of every effect constrain the values of its causes. For example suppose that there is the finding *level-of-consciousness = lethargy*, and that PHPT is the single-disease hypothesis being scored. In this case, the only node that can causally influence the *level-of-consciousness* node is the *serum-calcium-level*. The fact that the value of *level-of-consciousness* is *lethargy* implies that the *serum-calcium-level* must be between 10 and 18 mg/100ml (see Table 3-1). Such a constraint on the value of the *serum-calcium-level* can then be used to constrain the values of the nodes that cause it.

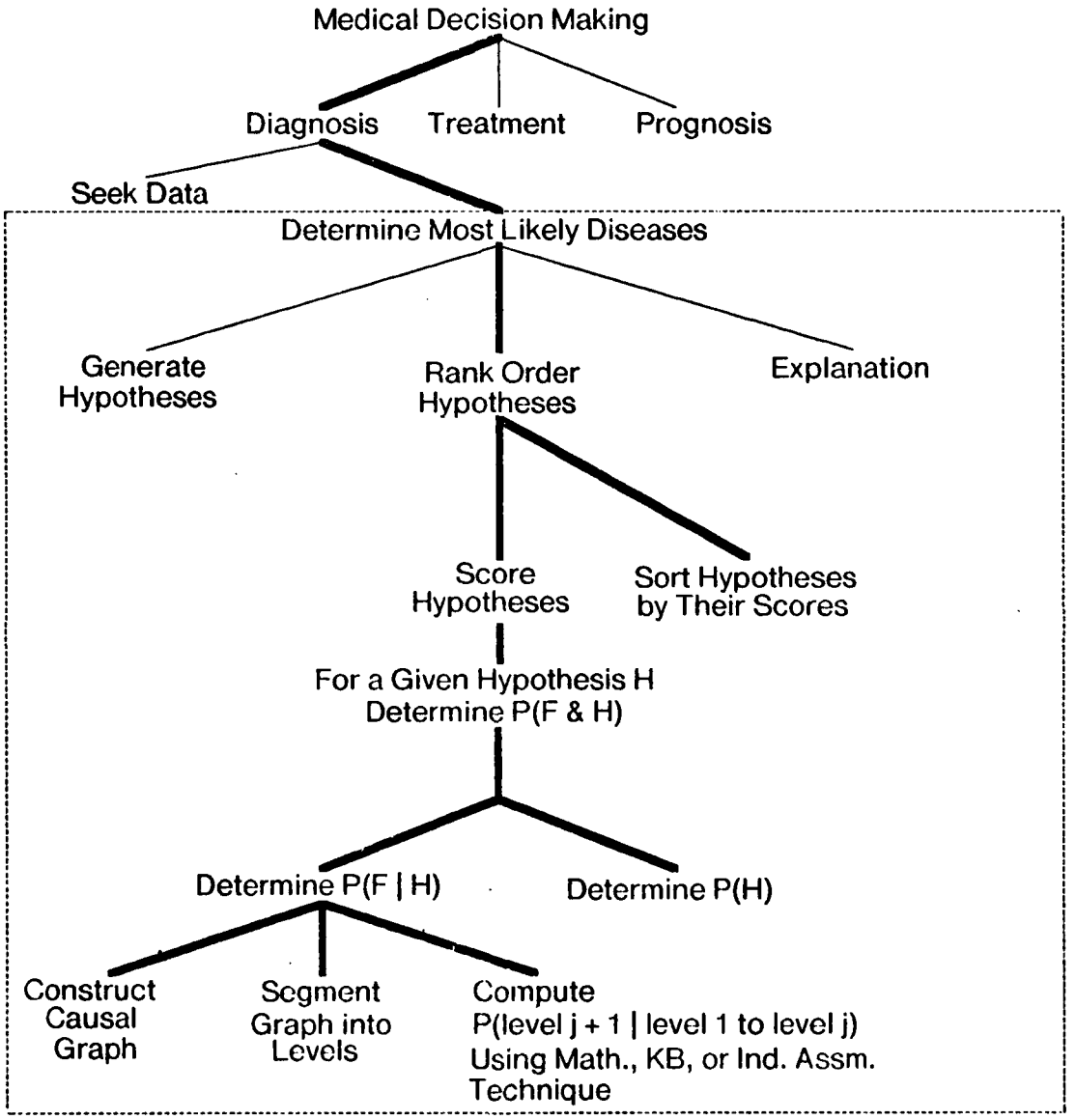


Figure 4-10: The Calculation of P(F | H) within the Context of Medical Decision Making

Altogether, such value constraints can lead to a substantial savings in the number of graph instantiations that must be considered.

4.3.2. Step2: Segment the Causal Graph into Levels

The purpose of this step is to place each of the nodes generated in step 1 into one of several possible levels in the graph. The algorithm classifies node x as belonging to level n if and only if n is the longest pathlength from any etiological node to node x. Figure 4-11 shows how a sample graph is segmented. Notice that a node can be causally affected by nodes from more than one level, as in the case of n_5.

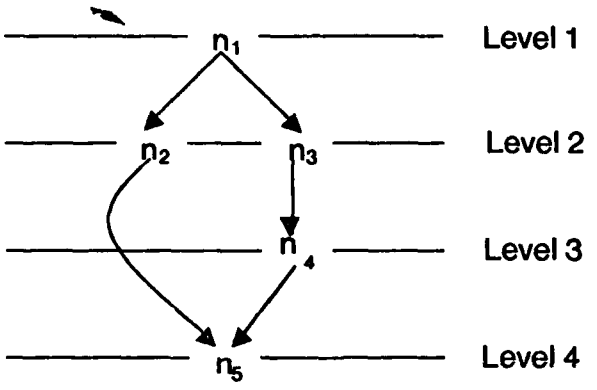


Figure 4-11: A Sample Segmentation of a Causal Graph

The important property of this ordering is that a node at level j can only be causally influenced by nodes at levels less than j. The levels are used in step 3 to order the sequence of calculations.

4.3.3. Step3: Compute the Score from the Segmented Causal Graph

The purpose of step 3 is to use the segmented causal graph from step 2 to compute the scoring metric SM = P(F & H) = P(F | H) x P(H). This is by far the most computationally expensive phase of the scoring process. The general equation on which it is based is as follows:

$$P(F \mid H) = \sum_{i_1=1}^{V_1} \cdots \sum_{i_N=1}^{V_N} \prod_{j=1}^{N-1} P(\text{values}(i_{j+1}, j+1) \mid \text{values}(i_1, 1) \,\&\ldots\&\, \text{values}(i_j, j))$$

where:

- N is the number of levels in the graph.
- V_k is the total number of possible unique value instantiations of (assignments to) the nodes at level k. These value instantiations are assumed to be ordered from 1 to V_k. For example if there are M binary variables at level 5, then $V_5 = 2^M$.
- values(i_j, j) is the i_j^{th} unique value instantiation of (assignment to) the nodes at level j.

P(values(i_{j+1}, j+1) | values(i_1, 1) & ... & values(i_j, j)) is the conditional probability of a set of values for the nodes at level j + 1 given a set of values for nodes at level 1 to level j. If this conditional probability could always be computed precisely, then the computation of SM would be straightforward. Unfortunately, it can not. NESTOR must use the typically sparse probability knowledge that *is* available, along with known causal knowledge and explicit assumptions, in order to bound the conditional probability as tightly as possible. The next section will explain how this is done.

4.3.3.1. Computing the Probability of Level j + 1 Given Level 1 to Level j

Figure 4-12 is an example of causal links between two levels of a larger causal graph. The single links between cause and effect nodes (variables) in the figure reflect the common situation in medicine in which only individual cause to effect conditional probabilities are available. That is, joint conditional probabilities are seldom available in the literature, and they are difficult for experts to estimate.

In the figure a node at level j + 1 may be causally influenced by nodes at levels 1 to j. Similarly, although not shown, the nodes at level j + 1 can causally influence other nodes at levels greater than j + 1.

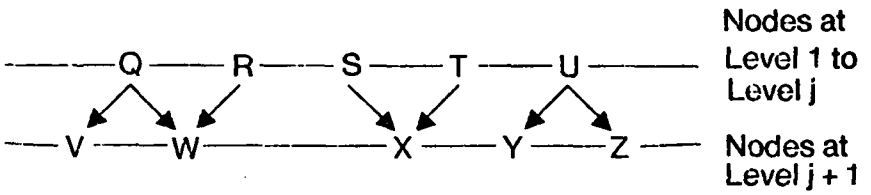


Figure 4-12: An Example of Causal Relationships Between Levels

The two links Q --> W and R --> W are called a ***convergent link group***. The links S --> X and T --> X form another such group. Figure 4-13 shows the conversion of convergent link groups to single links. At this point only single links and ***divergent link groups*** remain. Figure 4-14 shows the conversion of the divergent link groups in Figure 4-13 to single links. In this final form the composite links are independent of one another and thus their conditional probabilities may be multiplied together to derive the JCP of level j + 1 given levels 1 to j. which mathematically is expressed as follows:

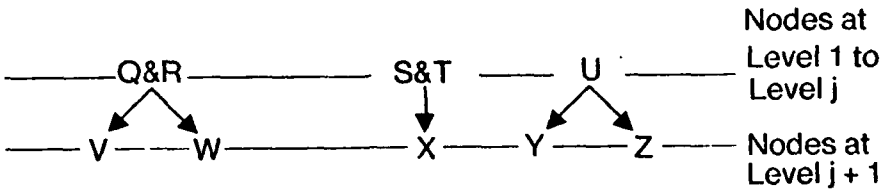


Figure 4-13: An Example of the Merger of Convergent Links

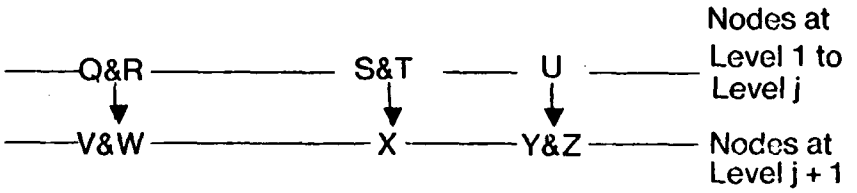


Figure 4-14: An Example of the Merger of Divergent Links

```
P(V&W&X&Y&Z | Q&R&S&T&U) =

P(V&W | Q&R) x P(X | S&T) x P(Y&Z | U)
```

This is the key calculation discussed in the previous section. Note that representing the relationship between levels in terms of convergent, divergent, and single link groups is sufficient to represent any acyclic causal graph. A method for handling cyclic graphs will be proposed Section 4.6.7. The next section, Section 4.3.3.1.1, shows how the transformation of convergent and divergent links to composite links is accomplished using knowledge about

the monotonic relationships between the cause and effect variables. The calculation for the example in Figure 4-12 will be used to demonstrate these transformations. This material will become detailed and technical at times. The main point to remember is that this is just one way in which the functional form of causal relationships (e.g., monotonicity) can be combined with sparse conditional probabilities and particular assumptions to derive bounds on a JCP. Sections 4.3.3.1.2 and 4.3.3.1.3 discuss two other methods of making this calculation which do not use causal knowledge as extensively as the method to be discussed next.

4.3.3.1.1 Method 1: The Use of Causal Knowledge

4.3.3.1.1.1 The Transformation of Convergent Link Groups

The goal at this stage is to merge the links in a convergent link group into a single link. An example is shown in Figure 4-15. Here three separate links, each specifying a conditional probability, are merged into a single link that represents a single conditional probability.

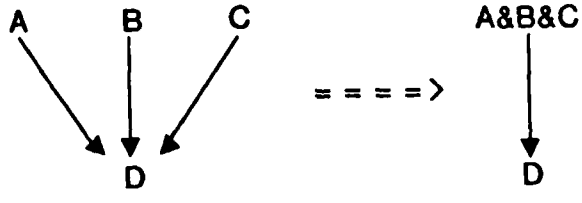


Figure 4-15: The Merger of Links in a Convergent Group

Every link has two pieces of information that are used in the merge process. First, there is the conditional probability in the top tier of NESTOR's causal model which has

already been mentioned. Second, there is ***monotonicity*** information which indicates something about the functional relationship between the cause and effect variables. An effect variable is assumed to be either a monotonically decreasing or monotonically increasing function of its cause variable.

The monotonicity knowledge is located in the bottom tier of NESTOR's causal model. The transformation of convergent links requires labeling the links in the top tier as either monotonically increasing or decreasing. In order to assign a monotonicity label to a link in the top tier, the bottom tier must be referenced. Suppose there is a link L that connects a cause C to an effect E in the top tier of NESTOR's causal model. If all the links in the bottom tier indicate a monotonically increasing relationship between C and E, then L is considered to be monotonically increasing also. An analogous inference establishes those links in the top tier that have a monotonically decreasing relationship between their cause and effect variables. Sometimes the bottom tier contains *both* increasing and decreasing monotonic relationships between C and E. In this case, NESTOR labels such links as having an *unknown* functional relationship. The convergent calculations then assume either increasing or decreasing monotonicity, always selecting at any given point the more conservative one (i.e., the one that maximizes the upper bound and minimizes the lower bound).

The assumption that functional relationships among variables can be characterized in terms of monotonicity is definitely a restriction of NESTOR's current causal representation. However, for the hypercalcemia domain it seems to be a reasonable assumption. It remains to be seen whether other domains within medicine will require a more complex repertoire of functional forms. In any event, the general point to be demonstrated is that knowing something about the functional form of the relationship between the cause and effect variables can aid in computing joint conditional probabilities from single conditional probabilities. That is, causal knowledge may supplement probabilistic knowledge. In particular, the next section will show how the monotonicity information is useful in merging convergent link groups.

In order to illustrate how links are merged, the simple case of two links converging on one node will first be developed, then generalized to any number of converging links. The general structure of the transformation is shown in Figure 4-16.

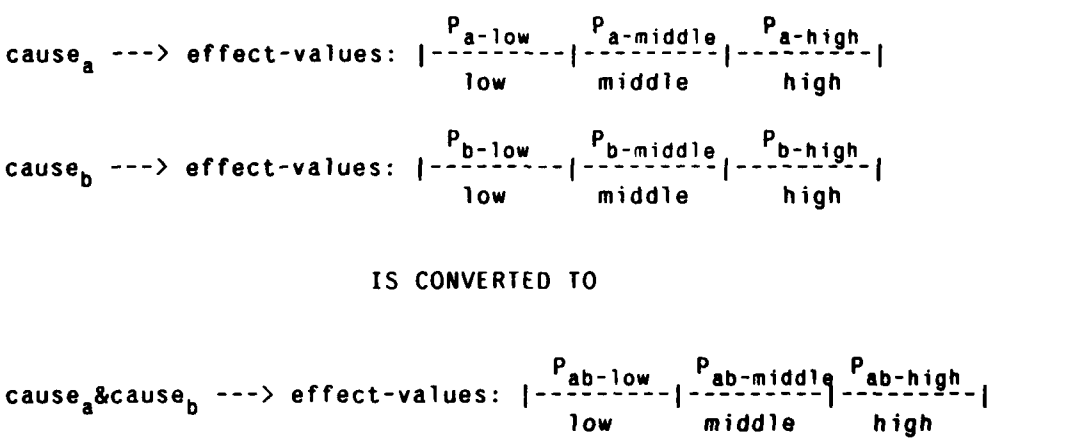


Figure 4-16: Structure of the Convergent Merge Transformation

Cause$_a$ and cause$_b$ are both nodes with instantiated values. In the Figure 4-16, cause$_a$ is shown as leading to three possible value ranges of the effect node, namely a low, middle, and high value. The low range is bounded below by the absolute minimum value the effect can attain. The high range is bounded above by the absolute maximum value. The middle range may be bounded above and below by any legal values of the effect. The probability of each of these three value ranges occurring is $P_{a\text{-}low}$, $P_{a\text{-}middle}$, and $P_{a\text{-}high}$, respectively; these three are actually probability *ranges*, since NESTOR bounds all probabilities. Any single range of values of the effect can be expressed as either a low, middle, or high range by appropriately defining the boundary points of one of them. Thus, the representation shown is general enough to relate any value range of a cause to any value range of an effect. In the figure the same points apply to cause$_b$. Also, Figure 4-16 shows the merged link at the bottom along with the probability for each value of the effect if both cause$_a$ and cause$_b$ are acting on it.

In making the transformation shown in Figure 4-16 there are three cases to be considered, based on the monotonicity of the links from cause$_a$ and cause$_b$ to the effect.

Case 1

Table 4-1 applies to the case in which the effect variable is an increasing monotonic function of both the $cause_a$ and the $cause_b$ variables. The table shows the upper and lower bounds of the probability of each value of the effect when both causes are acting on it. In the table entries, LB and UB refer to the lower bound and upper bound of a probability, respectively.

	Upper Bound Lower Bound
$P_{ab\text{-}low}$	$\min(UB(P_{a\text{-}low}), UB(P_{b\text{-}low}))$ $LB(P_{a\text{-}low}) \times LB(P_{b\text{-}low})$
$P_{ab\text{-}middle}$	$\max(0, 1 - LB(P_{ab\text{-}low}) - LB(P_{ab\text{-}high}))$ $\max(0, 1 - UB(P_{ab\text{-}low}) - UB(P_{ab\text{-}high}))$
$P_{ab\text{-}high}$	$1 - (1 - UB(P_{a\text{-}high})) \times (1 - UB(P_{b\text{-}high}))$ $\max(LB(P_{a\text{-}high}), LB(P_{b\text{-}high}))$

Table 4-1: Convergence Probabilities If Two Causes Increase the Effect

A major assumption in the derivation of the entries in the table is that the effect is a monotonically increasing function of $cause_a$ ($cause_b$), regardless of the value of $cause_b$ ($cause_a$); other assumptions are included in the discussion that follows concerning the derivation of the entries in Table 4-1.

Upper Bound of $P_{ab\text{-}low}$

Since each cause increases the effect, NESTOR assumes that both acting together will increase the effect at least as much as either acting alone. Thus, the likelihood that the effect will be low can be no greater than the likelihood that the strongest cause will leave it low. The strongest cause is the one that has the lowest probability of making the effect low, in other words, it causes a greater bulk of the probability distribution of the effect values to be in the middle and high ranges.

Lower Bound of $P_{ab\text{-}low}$

Since each cause is increasing the effect, it is theoretically possible that when both of them act together the probability of the effect being low is 0. This is certainly the most conservative lower bound possible. However, NESTOR assumes that the two causes do not act in such a strong synergistic manner on the effect. Instead, NESTOR assumes that any two causes that increase the effect will act at most independently to *decrease* the probability of a *low* value of the effect. This is equivalent to saying that NESTOR assumes that any two causes that increase the effect will act at most independently to *increase* the probability of a *non-low* value of the effect.

Upper Bound of $P_{ab\text{-}middle}$

This is expressed as a function of the lower bound of $P_{ab\text{-}low}$ (described above) and $P_{ab\text{-}high}$ (described below). It is based on the fact that the low, middle, and high values of the effect are exhaustive (i.e., the value of the effect must be one of them). Thus, the upper bound of a middle value of the effect is one minus the minimum value of the effect being low or high.

Lower Bound of $P_{ab\text{-}middle}$

This is derived in a similar manner to the upper bound of $P_{ab\text{-}middle}$.

Upper Bound of $P_{ab\text{-}high}$

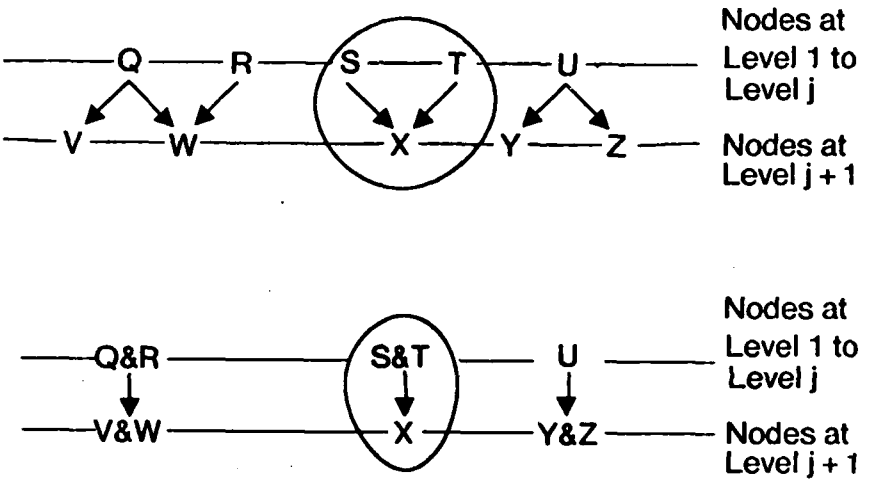
The two causes are assumed to act at most independently in causing the effect to be high. The reasoning is given above in the description of the lower bound of $P_{ab\text{-}low}$.

Lower Bound of $P_{ab\text{-}high}$

Since NESTOR assumes that each cause independently increases the value of the effect regardless of the other cause's value, both causes acting together will increase the effect at least as much as either acting alone. Thus, the likelihood that the effect will be high is at least as great as the likelihood that the strongest single cause will lead it to be high.

In Figure 4-17 an example of a convergent link group is circled. We will assume that the nodes S, T, and X all have the value *high*. The conditional probabilities of the two links in this group are shown in Figure 4-18. Since both links are monotonically increasing (as indicated by a plus sign), this is an instance of Case 1 just discussed. In particular the $P_{ab\text{-}high}$ entry in Table 4-1 is the relevant one, because the effect node X has a value which is

high. The upper bound assumes that S and T will at best act independently in causing X. This yields an upper bound of 1 - (1 - 0.4) x (1 - 0.7) = 0.82. The lower bound is taken as the maximum lower bound of the two causes, which in this example is 0.6. Thus, the bounded local JCP is as follows: $0.82 \geq P(X = \text{high} \mid S = \text{high} \,\&\, T = \text{high}) \geq 0.6$.



$$P(V\&W\&X\&Y\&Z \mid Q\&R\&S\&T\&U) = P(V\&W \mid Q\&R) \times P(X \mid S\&T) \times P(Y\&Z \mid U)$$

Figure 4-17: An Example Focusing on a Convergent Link Group

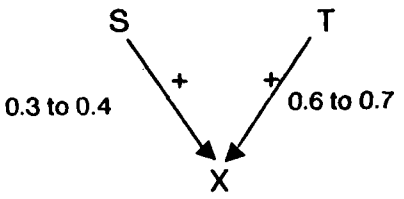


Figure 4-18: The Monotonicity and Conditional Probabilities of Convergent Links in the Example

Case 2

Table 4-2 is for the case in which the effect is a decreasing monotonic function of both $cause_a$ and $cause_b$. The entries in the table were derived in a manner analogous to those of Table 4-1. The only difference in the two cases is that the monotonicity of both links is reversed.

	Upper Bound Lower Bound
$P_{ab\text{-}low}$	$1 - (1 - UB(P_{a\text{-}low})) \times (1 - UB(P_{b\text{-}low}))$ $\max(LB(P_{a\text{-}low}), LB(P_{b\text{-}low}))$
$P_{ab\text{-}middle}$	$\max(0, 1 - LB(P_{ab\text{-}low}) - LB(P_{ab\text{-}high}))$ $\max(0, 1 - UB(P_{ab\text{-}low}) - UB(P_{ab\text{-}high}))$
$P_{ab\text{-}high}$	$\min(UB(P_{a\text{-}high}), UB(P_{b\text{-}high}))$ $LB(P_{a\text{-}high}) \times LB(P_{b\text{-}high})$

Table 4-2: Convergence Probabilities If Two Causes Decrease the Effect

Case 3

Table 4-3 is for the case in which the effect variable is an increasing monotonic function of the $cause_a$ variable and a decreasing monotonic function of the $cause_b$ variable.

	Upper Bound Lower Bound
$P_{ab\text{-}low}$	$UB(P_{b\text{-}low})$ $LB(P_{a\text{-}low})$
$P_{ab\text{-}middle}$	1.0 $min(LB(P_{a\text{-}middle}), LB(P_{b\text{-}middle}))$
$P_{ab\text{-}high}$	$UB(P_{a\text{-}high})$ $LB(P_{b\text{-}high})$

Table 4-3: Convergence Probabilities If One Cause Increases and Another Decreases the Effect

The entries in the table were derived as follows:

Upper Bound of $P_{ab\text{-}low}$

Since $cause_b$ tends to increase the likelihood of a low effect, while $cause_a$ *opposes* this, the likelihood of a low effect is assumed to be no greater than the upper bound of $P_{b\text{-}low}$.

Lower Bound of $P_{ab\text{-}low}$

Since $cause_a$ tends to decrease the likelihood of a low effect, while $cause_b$ *opposes* this, the likelihood of a low effect is assumed to be at least as great as the lower bound of $P_{a\text{-}low}$.

Upper Bound of $P_{ab\text{-}middle}$

Since the two causes are antagonistic, it is possible that when they are both in force the effect will always be in the middle value range. Thus, the probability is bounded above by 1.

Lower Bound of $P_{ab\text{-}middle}$

Since the two causes are antagonistic, there is a tendency for the probability distribution of the effect values to be "pulled" toward the middle value range. This is an assumption about how forces act in this domain. Thus, $P_{ab\text{-}middle}$ is assumed to be at least as probable as the minimum of the lower bound of $P_{a\text{-}middle}$ and the lower bound of $P_{b\text{-}middle}$.

Upper Bound of $P_{ab\text{-}high}$

Since $cause_a$ tends to increase the likelihood of a high effect, while $cause_b$ *opposes* this, the likelihood of a high effect is assumed to be no greater than the upper bound of $P_{a\text{-}high}$.

Lower Bound of $P_{ab\text{-}high}$

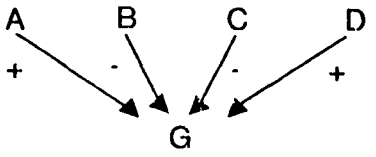
Since $cause_b$ tends to decrease the likelihood of a high effect, while $cause_a$ *opposes* this, the likelihood of a high effect is assumed to be at least as great the lower bound of $P_{b\text{-}high}$.

The three cases just presented can be specialized to deal with binary variables by removing the *middle* range entry. In this case, the value *low* becomes the value *false* and the value *high* becomes the value *true*. The formulas in the tables apply as before and for the same reasons.

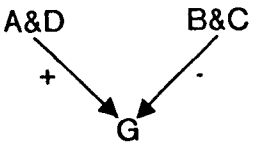
The above techniques have used knowledge of the functional form of the causal relationships (i.e., monotonicity) for merging two causal links. In the process, assumptions were adopted concerning the independence of causal links and their retention of monotonicity when acting together. The detailed, categorical knowledge that is available in the lower tier of NESTOR's causal model was not used to reason about when it is best to make independence assumptions, even though such reasoning is possible.

As we will see in the next section, the bulk of my research effort has been expended in developing the use of categorical knowledge for merging the more prevalent *divergent* link groups. The extension to reasoning more indepth about convergent link groups is an area for future research. However, even without these refinements, the above merging rules are conservative on the whole and would be expected to yield valid results in the current domain.

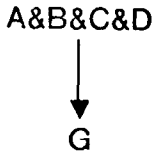
Thus far the techniques have dealt with how to merge only two convergent links to one link. However, in general there may be many convergent links as shown below:



The links are labeled with + or - depending on whether effect G is a monotonically increasing or decreasing function of its causal influences, namely A, B, C, and D. The method used to merge them into a single link first merges all + links. This is done by merging one + link with another to produce a combined link. Then, merging this composite link with another + link, and so on. This allows Table 4-1 to be used on successive pairs of links. The result is a single + link representing the merger of all the + links. Next, a single - link is created using the same process with Table 4-2. For the example above, this results in the following:



Finally, the + link and - link are merged using Table 4-3. Thus, the example becomes:



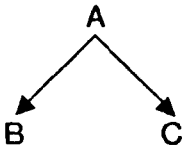
The main advantage to the convergent merge technique is that it is general. The main disadvantage is that it often results in wide bounds on the final conditional probability. One remedy for increasing validity and precision of the JCP of convergent links is to explicitly represent JCP's for the critical cases in which a frequent causal pathway contains a convergent link that either is computed very imprecisely (i.e., wide bounds on the JCP) using the tables above or does not meet the assumptions of either monotonicity or independence. As a case in point, when a convergent link group has one cause that increases the effect and another that decreases it, the above techniques (see Table 4-3) usually lead to wide bounds. In this case an explicit JCP would be very useful.

4.3.3.1.1.2 The Transformation of Divergent Link Groups

Figure 4-13 showed the causal relationships between two levels after all convergent links had been merged. What remains are divergent link groups.

A *divergent group* will be defined as the set of links that diverge from a single or composite causal node. The goal at this point is to compute for each group the joint conditional probability (JCP) of effect nodes of the group given the cause node. The JCP's of the divergent groups are then multiplied together to derive the probability of level $j+1$ given level j.

Suppose one group of divergent links looks as follows:



The goal is to compute P(B & C | A), where A, B, and C are each nodes with specific values assigned to them. Recall, that as a practical matter and an assumption of the implementation, NESTOR only has the conditional probabilities relating a single cause variable to a single effect variable, as for example P(B | A). Therefore, some method must be devised to compute the JCP from these individual conditional probabilities plus whatever causal knowledge is available.

There are two primary types of categorical causal knowledge that are available.

1. The causal direction of the relationship is known, namely A causes B, and A causes C. This is at the top of the causal model.

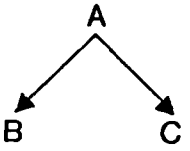
2. Detailed categorical causal knowledge of the monotonicity between the intermediate nodes linking A to B and A to C may be known. This detailed knowledge at the bottom level of the causal model can often be used to direct how to combine the individual conditional probabilities into a JCP.

Thus, the individual conditional probabilities and these two additional sources of knowledge are used to bound the JCP of a divergent link group. The method used to calculate the JCP depends on the nature of the detailed causal knowledge and the type of variables involved. Each method will be described below as a separate case with its own particular assumptions.

Case 1

Characterizing feature: There is no known detailed causal knowledge linking the cause to its effects

Description:



Here A is known to cause B and C at the top tier of NESTOR's causal model, but no detailed knowledge is available at the bottom causal tier to further refine the intermediate causal processes involved.

JCP Calculation:

$$\prod_j \text{UB}(P(\text{Effect}_j \mid \text{Cause}(s))) \geq \text{JCP} \geq \prod_j \text{LB}(P(\text{Effect}_j \mid \text{Cause}(s)))$$

where j ranges over the effects in the divergent group.

Assumptions:

If no detailed causal knowledge is explicitly available to interconnect variables such as B and C above, then they are assumed independent *given* their shared cause, which is A above. This is a form of default reasoning known as the closed world assumption: if you do not know something then assume it does not exist. The validity of this assumption clearly rests on how much is known about the causal mechanisms of the diseases being scored. This is why NESTOR is best applied to medical domains where the causal model is well understood. Although such default assumptions may be false at times and thus lead to error, they are preferable to CAMDM programs which uniformly assume conditional independence even when causal dependencies are known to exist.

There is one subtlety which should be noted. The value of the cause variable is assumed to be detailed enough so that knowing the value of any of the nodes it affects will not refine its value. If this were not true in the example above, then the value of B might refine the value of A which might affect the probability of C given A. In this situation B and C would clearly *not* be independent given A.

To make this point more concrete, suppose A is *serum calcium = high*, B is *level of consciousness = stupor*, and C is *muscle strength = normal*. Knowing that the patient is stuporous refines the value of the serum calcium by implying that it is *very* high. The fact that the serum calcium is high affects the probability that the muscle strength will be normal. In this case a very high calcium is much less likely to lead to normal muscle strength. Thus, knowing the value of one effect node (*level of consciousness = stupor*) has affected the probability of another node (*muscle strength = normal*), even though they are conditioned on the value of their sole cause. In this case it would be erroneous to calculate P(B & C | A) by assuming that P(B | A) and P(C | A) are independent, as done above.

In NESTOR this problem is avoided by insuring that the value of the cause variable is sufficiently refined. In the example above, the value of serum calcium would be some small range such as 14 to 15 mg/100ml. In this situation, knowing that the patient is stuporous can not by definition refine the serum calcium outside the narrow value range of 14 to 15, and knowing that the patient is stuporous does not significantly affect the probability of the muscle strength being normal within this *given calcium range*.

Case 2

Characterizing feature: The effect variables are binary with non-negative correlations

Description:

Recall that there is a two tier model in NESTOR. The top tier contains causal links with conditional probabilities that relate single cause nodes (variables) to single effect nodes. The bottom tier of the model contains only categorical relationships that connect the cause nodes to the effect nodes via intermediate nodes. The links at this lower level do not have probabilities associated with them. However, they do indicate whether the cause and effect variables have a monotonically decreasing or increasing relationship. Even though the bottom tier only contains qualitative information, it can be useful in calculating the JCP of a divergent link group.

The categorical links at the bottom tier of the causal model are used to determine if the effect variables can possibly be negatively correlated. Negative correlation means that if there is a cause A and two effects, B and C, then $P(B \& C \mid A) < P(B \mid A) \times P(C \mid A)$. That

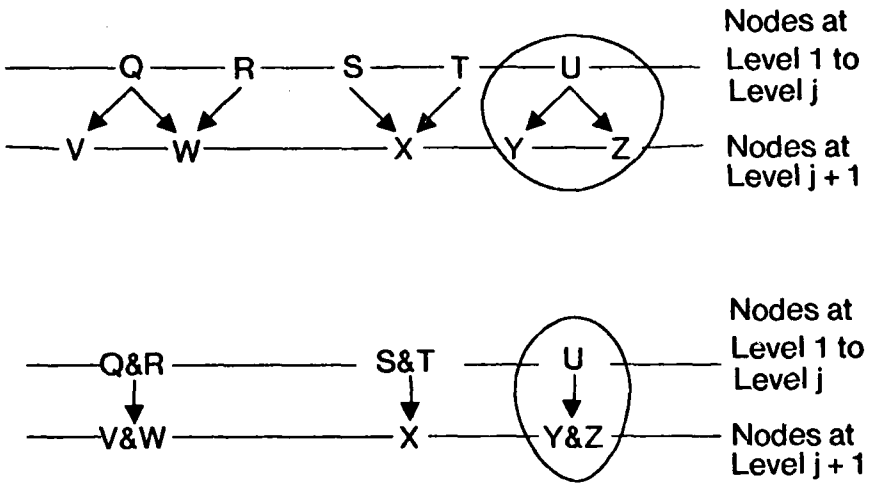
is, the JCP is less than if the effect variables are considered conditionally independent of one another. On the other hand, if no negative correlations are found, then the lower bound of the JCP of the given divergent group is calculated by assuming that the effects are conditionally independent of one another given the cause(s) of the divergent group.

The task is thus to determine if the effect nodes that diverge from a given cause node are negatively correlated. The technique is best illustrated by example. In Figure 4-19 a divergent link group is circled. We will assume that nodes U, Y, and Z are binary variables that all have the value *true*.

The graph on the left of Figure 4-20 shows only the top tier of the model; the conditional probabilities are 0.8 to 0.85 and 0.7 to 0.75 for the links. The graph on the right shows the bottom tier of the model which contains causal knowledge of the monotonicity of the relationships between U, Y, and Z. Note that I_1, I_2, and I_3 are intermediate nodes. All that is known at this level of the model is the monotonicity of the links and the interconnection of the nodes.

To determine if Y and Z are negatively correlated each pathway between them that does not pass through U must be traced. For this example the result is that all pathways between the two nodes contain monotonically increasing links. Since Y and Z both have the value *true* (which is defined as *increased*), and there is no evidence of one antagonizing the presence of the other, they are assumed to not be negatively correlated. This is because there is no evidence in the categorical causal model to suggest that they *are* negatively correlated; the absence of such knowledge is taken as evidence that no negative correlations exist, which is a form of default reasoning.

In the general case, intermediate nodes that link effects to each other in the bottom tier of NESTOR's causal model are used in determining whether two effects may possibly be negatively correlated. If there is some intermediate node which tends to produce the value of one effect node of a given divergent group while antagonizing the production of the value of another effect node in the group, then the two effects *may* be negatively correlated; NESTOR conservatively assumes that they are negatively correlated. Otherwise, the two effect nodes are assumed to not be negatively correlated. If no pair of nodes in a divergent



$$P(V\&W\&X\&Y\&Z \mid Q\&R\&S\&T\&U) = P(V\&W \mid Q\&R) \times P(X \mid S\&T) \times P(Y\&Z \mid U)$$

Figure 4-19: An Example Focusing on a Divergent Link Group

group is negatively correlated, then the group of effect nodes is considered to not be negatively correlated, given the cause node.

If the group of effect nodes are not negatively correlated, then the product of the lower bound of their individual conditional probabilities serves as a lower bound of their JCP. In

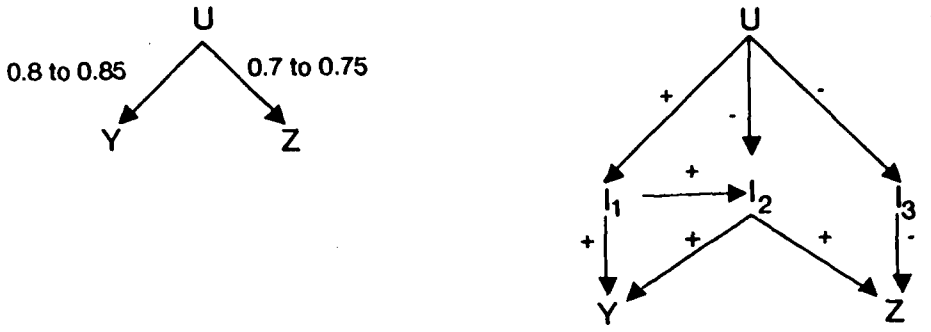


Figure 4-20: Causal Knowledge for a Divergent Group, Example 1

the example, Y and Z are not negatively correlated and thus the lower bound of P(Y = true & Z = true | U = true) is 0.8 x 0.7 = 0.56.

The upper bound is calculated by first clustering the links within a divergent group according to their independence. If two effect nodes are connected by *any* pathway in the bottom tier of the causal model (e.g., as Y and Z are connected) which does not include the cause node of the divergent group, then they are said to be connected. The transitive closure of connected effect nodes (within a given divergent group) partitions the top tier links of the divergent group into subsets. An arbitrary member of this partition is designated as $PART_i$. If two links are in different subsets of the partition, then they are assumed to have effect nodes that are conditionally independent of one another given the cause node of the divergent group. This assumption is another use of default reasoning in which effect nodes which do not have an interconnecting pathway in the bottom tier of the causal model (excluding pathways that pass through the cause node of the divergent group) are assumed to be conditionally independent.

The upper bound of a divergent group is calculated by using a mathematical property of probabilities. The maximum of the JCP of the effect nodes in some $PART_i$, given the

cause node of the divergent group, can be no greater than the smallest conditional probability of one of those effects; this is called $UB(P_{PART_i})$. Intuitively this is because a group of events (e.g., the causal relationships represented by the links in $PART_i$) can occur with no greater likelihood than the least likely event in the group.

Since the events in different subsets of the partition of a given divergent group are assumed to be conditionally independent, the product of $UB(P_{PART_i})$ over all the subsets i of the partition is an upper bound of the JCP of the divergent group.

In the example above there is only one partition, namely the set of links {U-->Y, U-->Z}. Thus, the upper bound of the JCP of the divergent group in Figure 4-20 is the (upper bound of the) *minimum conditional probability* among the two links, which is 0.75. If there had been a $PART_2$, then its links would connect additional effect nodes to node U. These effect nodes would be assumed to be conditionally independent of Y and Z given U. If $UB(P_{PART_2})$ were equal to 0.2, then the upper bound of the JCP of the divergent group would be 0.75 x 0.2 = 0.15.

Combining the upper and lower bound for the example in Figure 4-20 yields the following result: $0.75 \geq P(Y = true \ \& \ Z = true \mid U = true) \geq 0.56$.

JCP Calculation:

$$\prod_i UB(P_{PART_i}) \geq JCP \geq \prod_j P(Effect_j \mid Cause(s))$$

where i ranges over the subsets of the partition of the links in the divergent group, and j ranges over the effects in the divergent group.

Note that Case 1 is a special case of Case 2 in which every link in the divergent group is a separate subset in the partition of the divergent group.

Assumptions:

1. It is worth restating that NESTOR makes an assumption that all the important relationships between cause and effect nodes are represented in the bottom tier of its causal model. In this case, if there is an important link missing in the

bottom tier of the model, then a negative correlation may be overlooked and the JCP inaccurately calculated.

2. Another assumption concerns the nature of the relationships between cause and effect variables. The technique in this case depends on being able to characterize the monotonicity of the links. The precise meaning of this was covered in Chapter 3. The important point for now is that even though most relationships in the current domain can be described in terms of monotonicity, in other domains this representation may be inadequate. By stating the monotonic relationships as *unknown*, NESTOR can handle relationships in which either the functional relationship is not known or is not simply monotonic. However, this generally results in wide bounds on the probability of the hypothesis being scored.

Case 3

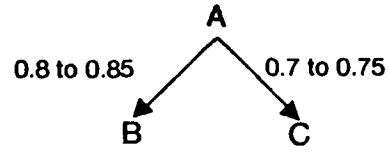
Characterizing feature: The effect variables are binary with some negative correlations among them

Description:

This is the situation in which there is at least one negative correlation among some pair of effect nodes. The upper bound calculation remains the same as in Case 2, however, the lower bound is calculated differently.

When there is a possibility of a negative correlation among the effect variables of a JCP, NESTOR calculates the lower bound of the JCP as though the negative correlation were as strong as possible. This is certainly a conservative measure, but it is necessary since the current representation only assumes the availability of qualitative knowledge of correlations.

In the case of negative correlations, the lower bound is calculated by considering the events represented by the negatively correlated effect variables to be maximally exclusive in terms of their occurrence. For example, consider the following case in which B and C are deduced to be negatively correlated:



We wish to calculate a lower bound for the following JCP: P(B & C | A). To understand how the lower bound is calculated, consider this as a bin packing problem. There are 100 bins. The i^{th} bin represents the i^{th} sample of the state of the world, which in this example is limited to the co-occurrence of events B and C. 100 samples of the world are made. In at least 80 of 100 samples B will occur because LB(P(B | A)) = 0.8 . So, a B type marker is placed in 80 bins. Similarly, 70 C markers are placed in 70 bins. Suppose that the world is such that the number of bins with both B and C markers is minimized. This corresponds to making the two events maximally exclusive of one another. Since there are 80 B type markers, there are 20 bins not occupied by B markers. Minimizing overlap means placing 20 of the 70 C markers in these 20 bins. This leaves 50 C type markers which must be placed in bins with B type markers. Thus, there are 50 of the 100 bins with *both* B and C type markers. Thus, LB(P(B & C | A)) $\geq$ 50%.

A generalization of this lower bounding procedure is as follows:

$$JCP \geq Maximum(0,\ 1 - \sum_{j} (1 - LB(P(Effect_j \mid Cause(s))))$$

where j ranges over all the effect variables in the divergent group.

This calculation is maximally conservative. By considering the conditional independence of subsets of effects, as done in Case 2 above, the lower bound can be increased (i.e., tightened). As in Case 2, an arbitrary subset of the partition of links in a divergent group is designated as $PART_i$.

A lower bound is calculated for each subset, and is designated as $LB(PART_i)$. The above inequality is used to calculate the JCP $LB(PART_i)$ for an arbitrary $PART_i$ if any two effects in $PART_i$ are indicated as possibly being negatively correlated by categorical causal knowledge in the bottom tier of NESTOR's causal model. Intermediate nodes that link

effects to each other in the bottom tier of NESTOR's causal model are used in determining whether two effects may be negatively correlated. Suppose there is some intermediate node which tends to produce one effect of a given divergent group while antagonizing the production of another. In this case the two effects may be negatively correlated. As an example, consider the divergent group shown in Figure 4-21.

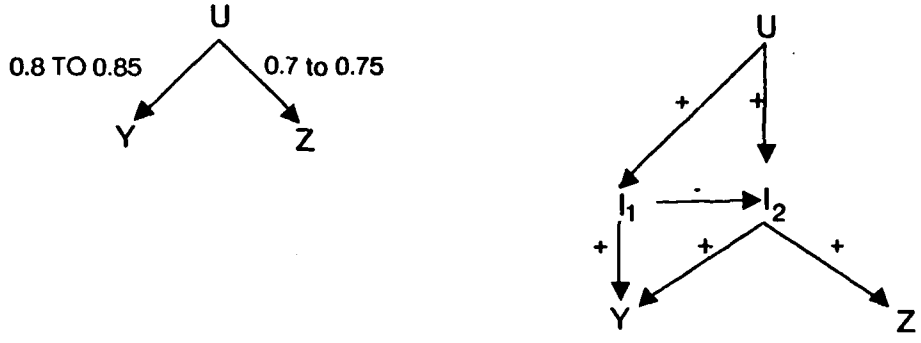


Figure 4-21: Causal Knowledge for Divergent Group, Example 2

In this case the effects $Y = true$ and $Z = true$ may be negatively correlated[34] given U, because $U = true$ tends to cause $I_1 = true$ which in turns tends to cause $Y = true$, while antagonizing the state in which $Z = true$. The context in which $P(Z = true \mid U = true) = 0.7$ to 0.75 is defined is free of any explicit consideration of the value of effect Y (or any other variable). Suppose the occurrence of $Y = true$ and $I_1 = true$ is very rare, and the usual pathway by which $U = true$ causes $Z = true$ is as follows: $U = true$ causes $I_2 = true$ causes $Z = true$. When $U = true$, this pathway causes $Z = true$ with

[34] Recall, that negative correlation is used to mean that $P(Y = true \ \& \ Z = true \mid U = true) < P(Y = true \mid U = true) \times P(Z = true \mid U = true)$.

probability 0.7 to 0.75. However, knowledge that $Y = true$ is significant because it suggests that $I_1 = true$, which suggests the possible inhibition of $I_2 = true$. The inhibition of $I_2 = true$ suggests that $Z = true$ may be inhibited. Thus, $P(Z = true \mid U = true \& Y = true)$ may be much less than 0.7. This implies that the JCP of the two effects given the cause may be much less than the product of their respective conditional probabilities, as shown by the following relationships:

```
   P(Y = true & Z = true | U = true)
=  P(Z = true | U = true & Y = true) x P(Y = true | U = true)
<  P(Z = true | U = true) x P(Y = true | U = true)
```

In this case, the inequality

$$JCP \geq Maximum(0,\ 1 - \sum_{j} (1 - LB(P(Effect_j \mid Cause(s)))))$$

is used to calculate the JCP $LB(P_{PART_i})$, where j ranges over the effect variables of the links of $PART_i$. This inequality assumes that the effects in $PART_i$ are maximally exclusive in terms of their co-occurrence. For this example, the result is $JCP \geq \max(0, 1 - ((1 - 0.8) + (1 - 0.7))) = 0.5$.

If none of the links in some $PART_i$ are negatively correlated by the above criteria, then the effects in $PART_i$ are assumed to co-occur (given the causes of the divergent group) with no less frequency than if they are considered independent. Thus, $LB(P_{PART_i})$ is calculated as follows:

$$JCP \geq \prod_{j} LB(P(Effect_j \mid Cause(s)))$$

where j ranges over the links in $PART_i$.

Finally, since the events in different subsets of the partition (i.e., different $PART_i$'s) are assumed to be conditionally independent, the product of $LB(P_{PART_i})$, for i ranging over every subset, is a lower bound of the JCP of the group.

JCP Calculation:

$$\prod_{i} UB(P_{PART_i}) \geq JCP \geq \prod_{i} LB(P_{PART_i})$$

where i ranges over the subsets of the partition of the divergent group as defined previously in Case 2.

Case 2 is a special case of this inequality in which no $PART_i$ contains effects which by the above criteria are negatively correlated.

Assumptions: The same as Case 2.

Case 4

Characterizing feature: The effect variables are continuous with value ranges that are bounded on *one* side

Description:

This is a generalization of Cases 2 and 3 above. An example of a continuous variable with a value range that is bounded on only one side is a serum calcium that is high. This implies that there is some lower bound on the value of the serum calcium, but no upper bound. The other case would be a variable bounded from above, as for example a low serum calcium.[35] The reason that these may be treated as binary variables is that bounding a variable on one side bifurcates its value range and thus it can be regarded as a binary variable.

JCP Calculation:

See Cases 2 and 3.

Assumptions:

See Cases 2 and 3.

Case 5

Characterizing feature: There are some effect variables that are continuous with values bounded on *both* sides

---

[35] Although technically a low serum calcium is bounded from below by zero, this is an absolute lower bound and therefore for the purposes of this discussion it will still be considered bounded on only one side.

Description:

Case 5 is a generalization of Case 4 in which some of the effect variables have a value range that is bounded from above and below. An example is a serum calcium = 14 to 15 mg/100ml. This can *not* be treated as a binary variable, since there are three value regions, not two. The only means of bounding the JCP in this case is to apply purely mathematical restrictions. The inequalities in Case 3 are used with the added provisions that the effects of the links in each $PART_i$ are considered to be maximally co-occurring in the calculation of $UB(P_{PART_i})$ and maximally exclusive in the calculation of $LB(P_{PART_i})$.

JCP Calculation:

$$\prod_i UB(P_{PART_i}) \geq JCP \geq \prod_i LB(P_{PART_i})$$

where i ranges over the subsets of the partition of the divergent group as defined in Case 2.

In calculating each $UB(P_{PART_i})$, the effects in $PART_i$ are considered to be non-negatively correlated (see Case 2); in calculating each $LB(P_{PART_i})$ the effects in $PART_i$ are considered to be negatively correlated (see Case 3).

Assumptions: The only assumption is that the bounds on the individual conditional probabilities are valid.

At this point one method has been described for converting convergent and divergent groups into composite links. Since these composite links have no known causal interactions, they are assumed conditionally independent of one another, thus they may be multiplied together to derive the probability of level j + 1 given levels 1 to j. This is the key calculation in computing the probability of an instantiated graph. The sum of the probabilities of all instantiated graphs is equal to P(F | H). Figure 4-22 shows the results for the ongoing example in this chapter. The conditional probability bounds for each group are shown under the group.[36] The probability of P(V&W | Q&R) was not calculated in this chapter and its probability bounds are included for the sake of completeness. To calculate it, the links Q --> W and R --> W would be converged to Q & R --> W, using the convergent

---

[36]The probability P(X | S&T) was calculated on page 102. The probability P(Y&Z | U) was calculated on page 115.

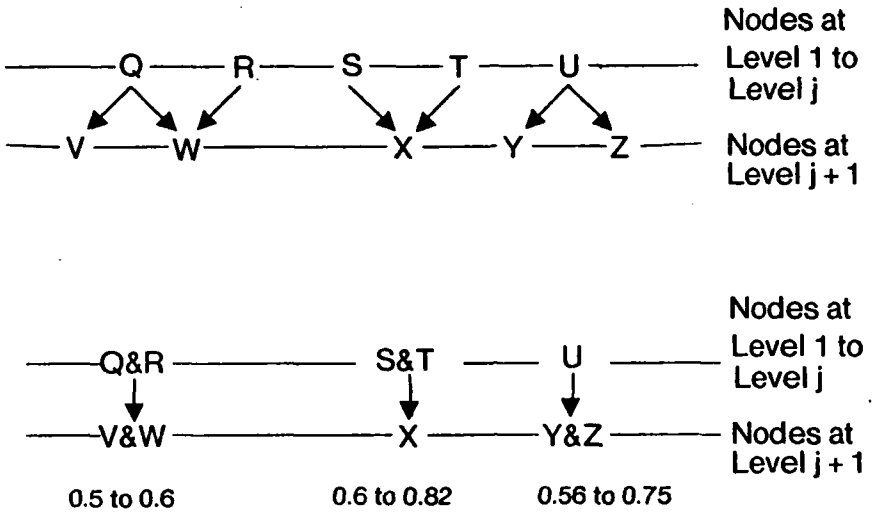


Figure 4-22: Summary of Local Joint Conditional Probability Calculations of an Example

methods discussed in Section 4.3.3.1.1.1. Next, the divergent links Q & R --> W and Q & R --> V (= Q --> V) would be converted to the composite link Q & R --> W & V, using the divergent methods discussed in Section 4.3.3.1.1.2.] Since the groups are independent of each other, the product of their probabilities is the probability of their joint occurrence. Thus, the probability of level j + 1 given level 1 to level j is as follows:

```
P(V&W&X&Y&Z | Q&R&S&T&U)
= P(V&W | Q&R) x P(X | S&T) x P(Y&Z | U)
= (0.5 to 0.6) x (0.6 x 0.82) x (0.56 to 0.75)
= 0.5 x 0.6 x 0.56  to  0.6 x 0.82 x 0.75
= 0.16 to 0.37
```

There are two other methods that NESTOR can use to calculate the probability of

some level j + 1 given level 1 to level j. These are discussed in the next two sections.[37]

4.3.3.1.2 Method 2: The Use of Independence Assumptions

Method 2 uses the conditional probabilities in the top tier of NESTOR's causal model, but does not use the categorical causal knowledge in the bottom tier. Instead the links are considered to be independent of each other.

4.3.3.1.2.1 Convergent Links

The JCP of a convergent link group is calculated by assuming that the links converging on an effect node are acting independently to produce the value of that node. For example, suppose that *A = increased* causes *B = increased* with probability 0.5, and that *C = increased* causes *B = increased* with probability 0.6. Using NESTOR's independence assumption results in the following convergent group probability:

```
P(B=increased | A=increased & C=increased)
= 1 - [1 - P(B=increased | A=increased)] x [1 - P(B=increased | C=increased)]
= 1 - [1 - 0.5] x [1 - 0.6] = 0.8
```

Note that this is a special kind of independence assumption, not to be confused with *conditional independence*, which states that A and B are conditionally independent given C if P(A & B | C) = P(A | C) x P(B | C).

JCP Calculation

$$1 - \prod_j (1 - UB(P(Effect \mid Cause_j)))$$

$$\geq \quad JCP \quad \geq$$

$$1 - \prod_j (1 - LB(P(Effect \mid Cause_j)))$$

where j ranges over the causes in the convergent group.

Assumptions

The major assumption, of course, is that the causes of the convergent group act independently to produce the value of the effect. In addition, this method ignores the

---

[37] Chapter 2 discusses how the user can select a particular method, and Chapter 5 discusses how NESTOR automatically selects a method when searching for the most probable diagnostic hypothesis.

functional nature of the individual causal relationships, and this may lead to bounds on the JCP that are invalid. The method has the greatest likelihood of deriving valid bounds on the JCP if 1) the effect is a binary variable, 2) the effect has a known value, and 3) the value of each cause variable independently increases the likelihood that the effect variable will have its current value.

4.3.3.1.2.2 Divergent Links

Divergent links are calculated by assuming conditional independence of the effects.

JCP Calculation

$$\prod_j UB(P(Effect_j \mid Cause(s)))$$

$$\geq \quad JCP \quad \geq$$

$$\prod_j LB(P(Effect_j \mid Cause(s)))$$

where j ranges over the effects in the divergent group.

Assumptions

This method assumes that the effects are conditionally independent given the cause node of the divergent group.

The calculation of a level $j+1$ given level 1 to level j is the same as with method 1, except that the convergent and divergent techniques of method 2 are used.

4.3.3.1.3 Method 3: The Use of a Mathematical Bounding Technique

It is possible to bound a JCP corresponding to a group of links without making any assumptions, other than the obvious assumption that the bounds on the individual conditional probabilities are valid. In particular, Case 3 of Section 4.3.3.1.1.2 introduced a lower bounding technique, and Case 2 introduced an upper bounding technique. These techniques are used in a slightly modified form below to bound the JCP of groups of links, such as the three in Figure 4.22.

JCP Calculation

The JCP for a group of links is bounded as follows:

$$\text{Minimum}(1,\ \underset{j}{\text{Minimum}}(\text{UB}(\text{Divergent-Link}_j)))$$

$$\geq \quad \text{JCP} \quad \geq$$

$$\text{Maximum}(0,\ 1 - \sum_{k} (1 - \text{LB}(\text{Divergent-Link}_k)))$$

where j and k range over every link the group.

This calculation assumes that all convergent link groups have been converted into composite links. Suppose that links Q-->W and R-->W, which form a convergent group in Figure 4-12, are converted to the composite link called Q&R-->W. The link Q&R-->W is used as a Divergent-Link in the above calculation. The bounds on the conditional probability associated with it are determined by assuming the trivial upper bound of 1.0 and the trivial lower bound of 0.0, although the upper may be less than 1.0. Thus, this method makes no assumptions about the likelihood that multiple causal states will cause an effect state. The disadvantage of this conservative approach is that the bounds on the above JCP calculation can be wide. For example, if there is *any* convergent group, then the above calculation will have a lower bound of 0.0.

In Figure 4.14 the above inequality would be used to bound P(V&W | Q&R), P(X | S&T), and P(Y&Z | U). The probability of level j + 1 given level 1 to level j would as follows:

$$\text{UB}(P(V\&W \mid Q\&R)) \times \text{UB}(P(X \mid S\&T)) \times \text{UB}(P(Y\&Z \mid U))$$

$$\geq \quad P(V\&W\&X\&Y\&Z \mid Q\&R\&S\&T\&U) \quad \geq$$

$$\text{LB}(P(V\&W \mid Q\&R)) \times \text{LB}(P(X \mid S\&T)) \times \text{LB}(P(Y\&Z \mid U))$$

Assumptions

The individual conditional probabilities are assumed valid. If two variables are causally related, then NESTOR assumes that there is a probabilistic causal link between them in its knowledge-base.

The ability to calculate the probability of one level given previous levels allows NESTOR to compute the probability of an instantiated graph, since its probability is equal to the following:

```
P(level n | level 1 to n-1) x ...
   x P(level j+1 | level 1 to level j) x ...x P(level 1)
```

Earlier it was shown that the sum of all possible instantiated graphs equals P(F | H), which is the probability of the current set of findings given the current hypothesis. Thus, the detailed level to level calculations that have been discussed are fundamental to calculating P(F | H). Since the scoring metric, P(F & H), equals P(F | H) x P(H), what remains is to calculate P(H).

4.3.3.2. Calculating P(H)

The etiologies of the hypercalcemia-causing diseases currently represented in NESTOR are independent of one another, so the calculation of P(H) is just the product of the individual prior disease-etiology probabilities. However, in other domains the etiologies may be probabilistically dependent. In such a situation NESTOR needs a method for bounding the joint probability of the disease etiologies. Fortunately, the techniques for calculating P(F | H) can be applied to calculating P(H), as long as the causal relationships between the disease etiologies are known.

Figure 4-23 shows an example of the causal relationships between the etiologies of the five diseases that comprise some hypothesis H. The goal is to compute P(H) = $P(E_a \& E_b \& E_c \& E_d \& E_e) = P(D_a \& D_b \& D_c \& D_d \& D_e)$. The subgoal is to compute the probability of each connected graph, which in the example is $P(E_a \& E_b)$ and $P(E_c \& E_d \& E_e)$. For each graph, the etiologies without a cause are treated as ultimate etiologies. Thus, E_a and E_c are ultimate etiologies. The next step is to compute P(non-ultimate etiologies | ultimate etiologies). This can be done using the same techniques that were used to compute P(Findings | Hypothesis). Here, etiologies such as E_b, E_d, and E_e are being treated like findings. Next, the score of each graph is computed as

```
P(non-ultimate etiologies | ultimate etiologies) x P(ultimate etiologies)
```

, where P(ultimate etiologies) is just the product of the prior probabilities of the ultimate etiologies in any subgraph. Once the probability of each causal subgraph has been

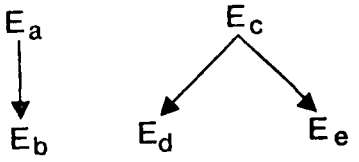


Figure 4-23: The Causal Relationships Among Five Hypothetical Diseases

calculated in this manner, they may be multiplied together (since they are independent) to derive the joint probability of all the disease-etiologies co-occurring.

The meaning of P(Etiology) for some disease D is important to the interpretation of any P(H), where D is a member of hypothesis H. The meaning of P(Etiology) is more accurately stated as P(Etiology | Noncausal findings). Noncausal findings include for example demographic information such as age and sex. So, in NESTOR it is through this conditional probability that the influence of noncausal findings express their affect on the score of an hypothesis. Currently in NESTOR the calculation of P(F | H) is not directly influenced by noncausal information. Theoretically, it could be, but this would require defining a set of conditional probabilities for every important combination of noncausal findings. At first, this may seem to imply that NESTOR assumes the conditional probabilities of causal links to be independent of noncausal findings such as age. However, this is not necessarily the case, since NESTOR can *bound* a conditional probability such that it is valid for *any* given combination of noncausal findings. In some cases, however, this may lead to wide bounds on the probability.

4.4. Related Work

This section reviews previous research concerning diagnostic programs that use a causal model. Table 4-4 outlines the major points of comparison. The features being compared are first outlined, then each system is discussed.

4.4.1. A Description of the Features Being Compared

Score is a formal probability

Each program in Table 4-4 that uses uncertain knowledge expresses the score of a diagnostic hypothesis in terms of a number. In some of the programs this is a formal probability, while in others it is an ad hoc number which usually bears some resemblance to a probability given particular assumptions.

Scoring metric

This describes the meaning of the scoring metric, regardless of whether it is a formal probability or ad hoc.

Links have formal probabilities

This indicates whether the weight expressed in a causal link is a formal conditional probability.

Probabilities are bounded

This indicates whether the probabilities in the knowledge-base and the score of diagnostic hypotheses can be expressed with bounds or if they must be a single, exact number.

Type of variables

The type of variable representing a node in a causal graph can be either binary, multivalued, or continuous.

Categorical causal knowledge

There is a distinct class of causal knowledge that either occurs with certainty or does not need certainty information in order to be used.

	Rous. 1	Lemm. 2	Quin. 3	Weiss 4	Ludw. 5	Patil 6	Pople 7	Davis 8	NESTOR 9
Score is a formal probability	+	+	+	-	-	-	-	-	+
Scoring metric	H|F	H|F	H|F	{H|F}	~H|F	Heur.	Heur.	Cat.	H&F
Links have formal probabilities	+	+	+	+	+	-	-	-	+
Probabilities are bounded	-	-	+	-	-	NA	NA	NA	+
Type of variables	B	B	B	B	B,M	B,M,C	B	B	B,M
Categorical causal knowledge	-	-	-	-	-	+	+	+	+
Probabilistic causal knowledge	+	+	+	+	+	-	-	-	+
Causal hierarchy	-	-	-	-	-	+	+	+	+
Diagnostic hierarchy	+	+	+	-	-	-	+	-	-
Type of probabilistic assumptions used	Ind.	Min. Dis.	None	Ind.	Ind.	NA	NA	NA	Var.

Abbreviations: + = present, - = absent, NA = not applicable,
H&F = P(Hypothesis & Findings),
H|F = P(Hypothesis | Findings),
{H|F} = ad hoc score resembling P(Hypothesis | Findings)
~H|F = approaches P(Hypothesis | Findings) under certain conditions
Heur. = Heuristic score
Cat. = Categorical score based on logical consistency
B = binary, M = multivalued, C = continuous,
Var. = variable, Ind. = conditional independence,
Min. Dis. = minimum discrimination

Table 4-4: Comparison of Diagnostic Programs that Use a Causal Model

Probabilistic causal knowledge

This indicates whether there are causal links which are qualified by conditional probabilities.

Causal hierarchy

A causal hierarchy exists if a link or node of a causal graph can be defined in terms of a causal subgraph.

Diagnostic hierarchy

A diagnostic hierarchy allows nodes in the causal graph to be grouped according to some logical relationship (such as disjunction in the case of binary valued nodes). Generally, this is used to construct a static tree of diagnostic categories.

Type of probabilistic assumptions used

This indicates the type of assumptions used in computing joint conditional probabilities within a causal graph.

4.4.2. Comments on Each Program

1. Rousseau

This relatively unknown piece of early work addressed many of the problems associated with probabilistic causal modeling [Rousseau 68]. Its main feature is the use of probabilistic causal knowledge to score diagnostic hypotheses. The program was restricted to binary variables and uniformly assumed local conditional independence when computing JCP's. *Local independence* means that the *direct* effects of a causal node are assumed to be independent given the cause. This is significantly more conservative than the more radical assumption of most Bayesian programs in which findings are considered independent given the general context of the disease being scored. The most important accomplishments of this work are that it developed the basic formal structure for scoring an hypothesis using causal knowledge, and it pointed out a number of areas for future research.

2. Lemmer

Lemmer's program is conceptually similar to Rousseau's in structuring the probability space with causal knowledge [Lemmer 76, Lemmer 82]. One major difference is that it can represent joint conditional probabilities and use them in scoring. This allows direct expression of probabilistic dependencies between causal variables when they are known. Another difference is that it uses a method to calculate joint probabilities when they are not explicitly known. Instead of maintaining bounds on a probability, this method makes a particular assumption about the distribution between those bounds and then derives a single number. Konolige later explored a similar approach to the same problem by using a maximum entropy criterion in calculating joint conditional probabilities [Konolige 79]. The problem with these methods is that they assume a distribution which may be incorrect and thus lead to an incorrect score. Banking on a single number increases the precision of the result, but increases the likelihood that it will be invalid.

3. Quinlan

This program, called INFERNO, is a very general probabilistic inference engine [Quinlan82]. It makes no assumptions *at all* about probability distributions or conditional independence. It constrains the bounds on nodes in a graph solely by use of sound mathematical methods. This approach leads to hypothesis scores that are valid, but also often very imprecise. It is also not able to use a priori probabilities. An additional problem with the approach is that it is not able to use categorical knowledge to make reasonable assumptions about the dependency of events. Also, it propagates probabilities within the graph in a local manner, and with no sense of the overall structure it may miss opportunities to significantly constrain the bounds of the nodes in the graph. It is so atuned to accurate local propagation of probabilistic constraints that is misses opportunities to pursue the bigger goal of distinguishing between diagnostic hypotheses. These weaknesses lead INFERNO to infer very conservative results (i.e., wide bounds), that may be of limited usefulness in a realistic medical domain. Nevertheless, INFERNO is an important body of research. It has emphasized the importance of using a formal inference technique and probability bounding in order to achieve a valid diagnostic score.

4. Weiss, et. al.

The CASNET program was the earliest program within the artificial intelligence in medicine community to use a causal model in diagnosis [Weiss 78]. One primary feature distinguishing the program from those above is its use of a heuristic inference technique to derive an ad hoc score of an hypothesis. This technique can be viewed as a heuristic version of Rousseau's program in that it considers only a portion of the causal simulation space. Like Rousseau's program, CASNET uniformly assumes conditional independence. CASNET also has a logical mapping from the causal graph to a disease classification tree. Although this can be implemented within Rousseau's system, it was not proposed. Additionally, CASNET is unique in using the causal progression of nodes within a disease graph to hypothesize the progression of the disease itself. Although therapy is not an area of primary concern in this comparison of programs, it is worth mentioning that CASNET did use its diagnostic assessment to determine a therapy. Although the diagnostic component of CASNET is in large part an heuristic adaptation of earlier methods, it has had a very positive impact because of its empirical demonstration of the importance of causal knowledge for a real-world medical diagnostic task.

5. Ludwig

This program is called INFERNET [Ludwig 81, Ludwig 83] and like CASNET it is a heuristic version of the program developed by Rousseau. However, the origin of its heuristic nature is considerably different from CASNET. Like CASNET, INFERNET only considers a subset of the causal simulation space when scoring an hypothesis, unlike Rousseau and Lemmer's programs which consider the entire space. It also assumes that effects are independent of each other given their immediate causes. However, unlike CASNET, the more simulation space INFERNET considers, the closer its scoring metric approaches a true probability. CASNET has a heuristic scoring scheme that does not produce a formal probabilistic score, regardless of how much of the simulation space is considered. The reason that INFERNET is heuristic is that it uses heuristics to decide which portion of the simulation space to consider. The problem with this approach is that the program is not able to detect when its heuristics (and attendant assumptions) are invalid. However, the notion of limiting the simulation space is an interesting one. Perhaps similar

methods can be developed to limit the simulation space while still guaranteeing the score's accuracy within specified bounds.

6. Patil

Patil developed a program called ABEL which uses a causal model to diagnose acid-base disorders [Patil81]. A particularly noteworthy feature of the program is its extensive use of a multilevel causal hierarchy in scoring a diagnosis. ABEL is also able to reason using mathematical models of causal interaction. However, the program uses no probabilistic knowledge in scoring diagnostic hypotheses.

7. Pople

Pople has proposed a new diagnostic program which combines a nosological hierarchy with a causal hierarchy [Pople82]. The formulation contains a general method for organizing a complex causal knowledge base and using it effectively for the generation of complex multiple-disease hypotheses. There is no specific method developed for scoring the hypotheses that are generated, but the representation ideas in this work are so interesting that it is included in this discussion. The method currently considers only categorical causal models. Unfortunately, the number of hypotheses that may be categorically consistent with a given set of findings may be very large. Thus, the extension of the representation and inference procedures to include probabilistic causal knowledge is an interesting and potentially fruitful area for further research.

8. Davis

This program [Davis 83] is representative of recent work concerning the diagnosis of electronic circuit faults using causal reasoning. Since the design of the circuit is completely known, the causal simulation is completely categorical (although it seems that using a priori probabilities could be potentially useful in hypothesizing the most likely faults to explore first). The program performs diagnosis by simulating how the circuit *should* behave, then contrasting this with how it *actually* behaves in order to narrow the possible location of the fault. Like ABEL, the program relies heavily on the use of a causal hierarchy in managing

exploration of a very large hypothesis space. A particularly interesting feature of the program is its use of multiple representations. In some cases, using a physical representation of components is best, while at other times a functional representation is best. The choice seems to depend on which representation allows the causal interaction of components to be viewed as a local interaction. Such multiple representations could also be useful in future medical diagnostic systems.

Other Related Research

In addition to the programs in Table 4-4, there is other research that is related to the concepts and techniques in NESTOR.

NESTOR uses P(F & H) as a scoring metric; the sufficiency of P(F & H) as a scoring metric to rank order hypotheses by their posterior probabilities has also been noted by Charniak [Charniak 83]. He too has realized relevance of using causal knowledge to structure the aggregation of probabilistic knowledge. The method he describes uses binary variables with conditional probabilities between variables that are represented with single numbers, rather than bounds. Local joint conditional probabilities (local JCP's) are calculated using either explicitly stored JCP's or are calculated assuming local conditional independence.[38] No technique is offered which can circumvent the need for causal simulation.

Pearl has developed a method which uses a modified form of Bayes' formula to update the probabilities of multivalued nodes in a tree-structured graph [Pearl 82]. The assumption of a tree-structured graph is a major practical limitation of this method, although in theory it can be avoided at the expense of specifying a very large number of JCP's. In propagating information through the tree, local JCP's are calculated by assuming local conditional independence.

Rowe has developed a program which infers upper and lower bounds on statistical

---

[38]Recall that this means that if A causes B and A causes C, then P(B & C | A) is assumed to equal P(B | A) x P(C | A).

parameters such as means and medians by using partial information about the complete population [Rowe 83]. In a like manner, NESTOR infers upper and lower bounds on a particular type of parameter, namely a conditional probability, by using a subset of the probabilities from the complete joint conditional probability space of the patient population.

4.4.3. A Comparison of NESTOR to Previous Research

A portion of NESTOR's design can be viewed as a synthesis of the previous programs discussed above. Like Lemmer and Rousseau's programs, it considers the complete causal simulation space in order to calculate a formal probability score for an hypothesis. Like Quinlan's INFERNO, it uses bounds on probabilities in order to maintain accuracy when not enough probability knowledge is available to compute an exact probability. Like Patil's ABEL, it uses categorical knowledge and multivalued causal variables in its causal reasoning.

NESTOR is unique among current CAMDM programs in the way it uses categorical knowledge to make informed assumptions about the local independence of probabilistic, causal interactions. Also, to my knowledge, it is the only running CAMDM program that uses P(H&F) as a scoring metric. This allows it to determine the most likely diagnostic hypotheses without having to explore the entire hypothesis space. This can lead to significant savings in computation time, and can be the difference between a computationally feasible and a computationally intractable approach.

The most unique aspect of NESTOR is the way it combines all these features into a single program. Of particular note is the way it uses a causal simulation scoring method in order to maintain a ***formal probabilistic score*** (like Lemmer and Rousseau's programs), but when faced with inadequate probability knowledge it examines known ***categorical relationships*** (such as appear in ABEL) in order to determine reasonable ***assumptions*** that can be employed in order to ***bound the probability*** (like INFERNO), thus ultimately yielding a score that is less precise, but still valid.

4.4.4. Features Lacking in All the Programs

There are several features that seem to be important in diagnosis, but which none of the above programs, including NESTOR, currently possess. One of the most conspicuous is the lack of an explicit, general representation of the temporal relationship between a cause and effect. As will be discussed in Section 4.6.6, handling this in a general manner is difficult. However, temporal reasoning is often crucial in diagnosis, thus it is important to begin considering how to extend the representations of diagnostic programs to handle temporal links explicitly and generally.

Each of the reviewed programs has its own particular built-in assumptions about the form of probabilistic and causal knowledge it will use and how it will use them. Most of the programs assume that probabilistic inferenceing will proceed from the cause to the effect. INFERNO is the least committed to this form of reasoning, although its representation does not allow a general conditional probability representation. There is a need for a program that allows *any* type probability to be expressed between *any* nodes, *regardless* of their causal relationship. It also needs to be able to represent a wide variety of causal models: structural, functional, mathematically detailed, qualitatively abstract, categorical, and probabilistic. The program then needs to be able to combine all this knowledge to maximally restrict the bounds on the score of the hypothesis it is scoring. Indeed such a general program will be difficult to develop, but current causal diagnostic programs contain many of the essential pieces with which to begin.

4.5. An Evaluation of NESTOR's Scoring Method

In order to assess the value of the causal knowledge-based techniques used by NESTOR, a study was conducted in which NESTOR's diagnostic accuracy on a set of hypercalcemia cases was compared to that of a Bayesian program that uniformly assumed conditional independence of the findings. The primary purpose of the study was to identify and analyze any instances in which the two methods differed in their diagnostic evaluation of the hypercalcemia cases.

4.5.1. Methods

4.5.1.1. The Two Computer Programs being Compared

The methods that NESTOR uses in scoring an hypothesis have already been covered in the previous sections of this chapter. In particular, it uses the causal structuring of findings and knowledge-based techniques for calculating local joint conditional probabilities. Preliminary analysis indicated that many of the hypotheses had overlapping scores, so that a definitive determination of the most probable hypothesis was not possible. For this reason I decided to use the upper bound of the score of an hypothesis in ranking the hypotheses. The implications of this decision were among the open questions of the study.

The Bayesian program (to be called BAYES) uses the typical assumption of conditional independence of the findings in calculating the score of an hypothesis. Since NESTOR's knowledge-base already contained a causal structuring of the findings, it was necessary to eliminate this structure in order to construct BAYES' knowledge base. For a given disease NESTOR was used to determine the probability that the disease would cause each of its findings independent of any other findings. So, if a disease had N findings, then NESTOR calculated P(f | D) for each finding f, and thus it determined N conditional probabilities relating the disease to each of its N findings. In calculating these probabilities NESTOR used the average value between the bounds of the conditional probabilities of intermediate causal links so that the final result was a single conditional probability, rather than a bounded probability; this is in keeping with the use of point probabilities such as are typically found in the knowledge-bases of Bayesian programs.

The scoring metric used by BAYES was the same as that used by NESTOR, namely P(F & H), where F is the set of case findings and H is the hypothesis being scored. P(F & H) is equal to P(F | H) x P(H). Since P(H) is available, the scoring process involves calculating P(F | H). BAYES uses two rules in making this calculation. For each finding f in F that is caused by more than one disease in H, it first converts the links converging onto f into a single composite link. To do this, it assumes that each of the diseases in H that affect f does so independently of the others. Thus, if D_1 and D_2 are the only diseases in H

influencing f, and D_1 causes f with probability p_1, and D_2 causes f with probability p_2, then $P(f \mid H) = p_1 + p_2 - p_1 \times p_2$. Once this has been calculated for each finding, the next step is to combine the conditional probabilities of each finding given the hypothesis into the joint conditional probability of all the findings given the hypothesis. Since BAYES assumes that the findings are conditionally independent, it multiplies all the individual conditionals together to derive the final joint conditional.

4.5.1.2. Case Classification

The hypotheses scored were those containing no more than two diseases. This corresponds to the case generation method discussed in the next section. in which no more than two diseases were used to generate a case. Since there were a total of seven individual diseases represented in the knowledge-bases of NESTOR and BAYES, each of these was scored. Additionally, each of the twenty-one double disease hypotheses was scored.

Every case was classified into one of three categories. If the score of all hypotheses was zero, then it was classified as *impossible.* This means that the diseases currently known to NESTOR are incapable of accounting for the findings in the case. If the upper bound of the score of the most probable hypothesis was less than two times the upper bound of the score of the second most probable hypothesis, then the case was classified as *insufficient data.* The ratio of two was chosen because it yielded reasonable results on preliminary trial cases. However, the exact value of the ratio is not critical since the primary purpose of the study is to compare NESTOR to BAYES, both of which used the same ratio. Finally, if neither of the above two classifications was made, then the most probable hypothesis was considered the best leading diagnosis that explained the findings in the case.

4.5.1.3. Case Generation

The following steps were taken in generating cases for the study:

1. A disease hypothesis was selected consisting of one or two diseases. The details of this step will be discussed later in this section. This hypothesis will be called H.

2. A number from 2 to 12 was randomly selected as the number of findings to be

included in the case; this number will be called N. [39]This range was chosen so that the performance of the two programs would be faced with a wide range of case information. On the basis of trial experiments a case with only two findings was likely to be best classified as containing *insufficient data*, whereas with twelve the likelihood of making a definitive diagnosis was high.

3. Next, N finding variables were randomly selected from among those that hypothesis H can causally affect. An elevated serum calcium level was always included as one of the finding variables.

4. An abnormal value was selected for each of the N finding variables. A finding variable plus its value will be called a finding. The probability of a finding variable being assigned a given value was equal to the probability of occurrence of that value of the finding in a patient with the diseases in hypothesis H. For example, the likelihood of the finding variable *serum calcium* being assigned a value of 10 to 12.5 mg/100ml was much greater than it being assigned a value of 19 to 20 mg/100ml because the former is much more likely to occur in the diseases being used for case generation. The causal interaction of the N findings was not considered in the generation of their values. This was done so that the robustness of NESTOR and BAYES could be measured with regard to detecting causally inconsistent findings. In practice, such robustness can be important in alerting the user that either the findings are in error or that the computer system does not have knowledge of the diseases that can cause this set of finding values.

5. If the N findings have already been generated in a previous case, then a return was made to step 2 and a different set of findings was generated. Otherwise, the N findings were considered as a new case.

In step 1 the hypothesis H was created in order to generate a set of case findings. Hypothesis generation was limited to one and two disease hypotheses because of the small number of diseases currently represented (i.e., seven) and the fact that the occurrence of three of them together is highly unlikely. Each of the seven diseases constituted a single disease hypothesis H. Cases were then generated using steps 2 to 5 given above until each single disease hypothesis had two cases generated using it in which NESTOR and BAYES disagreed on the diagnosis. In theory this would yield 7 x 2 = 14 cases in which NESTOR and BAYES

---

[39]The *total* number of findings in NESTOR's knowledge-base is about 70 (see Section 3.1.1).

disagreed. However, for practical reasons there was an upper limit of 40 placed on the number of cases to be tried using a given hypothesis; one hypothesis, non-metastatic cancer, reached this limit and only generated one case in which NESTOR and BAYES disagreed on the diagnosis. Thus, there were a total of 6 x 2 + 1 = 13 cases generated from single disease hypotheses in which NESTOR and BAYES disagreed.

Next, an equal number of cases (13) were generated from double disease hypotheses. Thirteen hypotheses were randomly selected from among the 21 possible double disease hypotheses. Steps 2 to 5 were used to generate for each hypothesis a single case in which NESTOR and BAYES disagreed on the diagnosis.

Thus, there were 13 + 13 = 26 cases generated in which NESTOR and BAYES disagreed on the diagnosis. Half of them were generated from single disease hypotheses and half from double disease hypotheses.

At this point a total of 188 cases had been generated in order to get the 26 cases in which there was disagreement. Cases were randomly selected from among all the 188 cases until there were 24 cases different from 26 already selected. These 24 cases were necessarily ones in which NESTOR and BAYES agreed on the diagnosis. A total of 27 samples were taken in order to get the 24 cases.[40] Thus, 3 of the 27 samples were cases in which NESTOR and BAYES disagreed on the diagnosis. The 26 cases of disagreement plus the 24 cases of agreement constituted the 50 cases that were used in the evaluation.

4.5.1.4. Case Evaluation

The order of the 50 cases was first randomized and then each case was printed in a standard format on a single sheet of paper. Figure 4-24 shows an example of a case sheet as presented to the expert who evaluated them. The top of the sheet contains a list of the seven possible diseases. Below this the findings in the case are listed.

The cases were judged by an endocrinologist, who will be called the *expert*. This

---

[40]The next section will clarify why this selection technique was used, rather than selecting 24 cases from among the remaining 188 - 24 = 164 cases.

expert was the same one who provided the primary *probabilistic* knowledge for five of NESTOR's seven diseases.[41] The use of one person as a knowledge-source and case-judge aided in controlling for misdiagnoses due to inter-expert knowledge-base differences. This is desirable, since the primary purpose of the study is to test the inference techniques and not the knowledge-base per se. However, it should be noted that the accuracy of BAYES and NESTOR, as measured in this study based on the judgement of one expert, will not be an independent test of the general clinical diagnostic accuracy of their knowledge-bases.

The expert was to first give his best assessment of the case. No time limit was imposed on him at this or subsequent stages of the evaluation. Next, he was to rate two assessments. The lower half of the case sheet contains the two assessments, which were covered by a piece of paper which was stapled into place. After removing the paper and examining the assessments, the expert rated each assessment and also rated which he considered to be more likely.

If a case was one of the 26 in which NESTOR and BAYES disagreed on the diagnosis, then one assessment corresponded to BAYES' diagnosis and the other to NESTOR's. If the case was one in which NESTOR and BAYES agreed on the diagnosis, then one assessment was that of the two programs and the other was a diagnosis randomly selected from one of the remaining 27 (= 28 - 1) single and double disease hypotheses. The assignment of the diagnoses to either assessment 1 or assessment 2 was balanced and randomized so that there would be no bias based on the order in which the expert read the assessments.

4.5.2. Results

The results are divided into two parts. First, the expert's evaluation of both BAYES and NESTOR is presented for the random sample of 27 cases. This indicates the individual accuracy of the programs based on the expert's assessment. Next, the expert's comparative evaluation of BAYES vs. NESTOR is given for the 26 cases in which the two programs

---

[41] The probabilities for the other two diseases were taken from a medical textbook [Perlroth 81].

```
DISEASE LIST:
     1. GRAVES DISEASE
     2. METASTATIC CANCER OTHER THAN MYELOMA
     3. NON-METASTATIC CANCER
     4. MYELOMA
     5. PRIMARY HYPERPARATHYROIDISM
     6. PHEOCHROMOCYTOMA
     7. SARCOIDOSIS
------------------------------------------------------------
CASE 1.

FINDINGS:
     TOTAL SERUM CALCIUM IS 16 TO 17 MG/100ML
     GENERAL MUSCLE STRENGTH IS WEAK
     LEVEL OF CONSCIOUSNESS IS LETHARGY
     CONSTIPATION IS PRESENT
     ANEMIA IS PRESENT
     BACK PAIN IS PRESENT
     FEVER IS PRESENT
     WEIGHT IS DECREASED
     WHITE BLOOD CELL COUNT > 11000 TOTAL/MM3


YOUR ASSESSMENT (SELECT ONE):

     ..... FINDINGS CANNOT BE EXPLAINED BY ANY COMBINATION OF
              ONE OR TWO DISEASES FROM THE DISEASE LIST ABOVE
     ..... INSUFFICIENT DATA TO SELECT A LEADING DIAGNOSIS TO
              EXPLAIN THE FINDINGS
         . BEST ASSESSMENT [DISEASE NUMBER(S)]: ..............


ASSESSMENT 1
     * METASTATIC CANCER OTHER THAN MYELOMA

     RATING (SELECT ONE):  ..... AN ACCEPTABLE ASSESSMENT
                           ..... A WEAK ASSESSMENT
                           ..... AN INCORRECT ASSESSMENT


ASSESSMENT 2
     * MYELOMA
     * SARCOIDOSIS

     RATING (SELECT ONE):  ..... AN ACCEPTABLE ASSESSMENT
                           ..... A WEAK ASSESSMENT
                           ..... AN INCORRECT ASSESSMENT


WHICH ASSESSMENT DO YOU CONSIDER MORE LIKELY:
     ..... ASSESSMENT 1
     ..... ASSESSMENT 2
     ..... BOTH EQUALLY LIKELY
     ..... BOTH EQUALLY UNLIKELY
```

Figure 4-24: An Example of a Case Evaluation Sheet

reached different diagnostic conclusions.[42]

4.5.2.1. Analysis of the Cases Generated by Random Sampling

As discussed Section 4.5.1, 27 of the 50 total cases that were evaluated were chosen by taking a random sample of all 188 of the cases originally generated. The performance of NESTOR and BAYES on these cases reflects how accurate the programs are in diagnosing cases in which there are one or two diseases from among the seven in their knowledge bases that are causing findings that are also currently represented in their knowledge bases.[43] Table 4-5 shows the results for the 27 cases. NESTOR and BAYES both had a rating of acceptable in 63% $\pm$ 9% of the cases (probability $\pm$ standard error). The similarity of the rating of NESTOR and BAYES is not surprising since 24 of the 27 cases were ones in which the two programs concluded the same diagnosis.

	Acceptable	Weak	Incorrect
NESTOR	17	9	1
BAYES	17	9	1

Table 4-5: The Ratings of the Cases Generated by Random Sampling

In order to understand the source of the errors, the diagnostic process used by NESTOR and BAYES was examined in detail for each case that received a weak or incorrect rating. Table 4-6 shows a tally of the error types.

The knowledge-based errors were due to inaccuracies or omissions in the knowledge base and were all correctable. For example, in one case the ability of Graves' disease to cause an increased white blood count was not represented. This erroneously led NESTOR to invoke another disease to account for this finding, when Graves' disease alone could account for all the findings.

---

[42] Note that three cases from the 27 case-sample occur in the 26 case-sample. Thus, there are a total of 50 cases that were evaluated by the expert.

[43] Again, it should be noted that in this study their accuracy is based on an expert's evaluation.

Type Error	NESTOR	BAYES
Knowledge-base Deficiency	3	0
Causal Reasoning Errors	0	3
Threshold Errors	7	7

Table 4-6: The Causes of Errors in the Cases Generated by Random Sampling

The three causal reasoning errors were due to BAYES' inability to consider the causal relationships of the findings in a case. Two of the errors were the way it incorrectly aggregates evidence. The other case was due to not recognizing the causal inconsistency of a set of findings. Examples of such errors will be given in the next section.

A threshold type error occurred when the following four conditions existed:

1. The expert assessed the case as one in which there was *insufficient data to select a leading hypothesis.*

2. NESTOR or BAYES did not assess the case as either containing *insufficient data to select a leading diagnosis* or as being *impossible.* That is, the diagnosis consisted of one or two specific diseases.

3. The the expert rated NESTOR or BAYES' assessment as weak or incorrect.

4. There was no other known knowledge-base or reasoning error which accounted for the error.

Recall, that NESTOR and BAYES rated a case as containing *insufficient data to select a leading hypothesis* only if the ratio of the upper bound of the top two hypotheses was less than two. A threshold of two was based on preliminary experiments in which it yielded reasonable assessments. However, it was not expected to coincide exactly to the expert's notion of insufficient data, and in seven cases it did not. Although these cases suggests that a more conservative threshold is needed, further experiments are necessary to determine its value more precisely.

The three knowledge-base errors reflected in Table 4-6 were not subtle and were thus easily corrected. The three causal reasoning errors were all due to inherent flaws in BAYES' scoring algorithm, and thus were not correctable without altering the definition of BAYES. The threshold type errors were potentially correctable by changing the ratio used. However, they were not corrected, because doing so would require the expert to reevaluate a number of cases that would be given assessments different from their original assessments. Testing this change would have essentially constituted the effort of a second experiment. Instead, to be maximally conservative, all seven threshold type errors were counted against NESTOR and BAYES. Since, the two programs had an equal number of threshold type errors, this did not constitute a bias in favor of one of them.

After eliminating the cases due to knowledge-based errors, NESTOR received an acceptable rating in all 17 of 24 cases (71% $\pm$ 9%), and BAYES received an acceptable rating in 17 out of 27 cases (63% $\pm$ 9%). This difference in performance is not very significant statistically ($P > 0.2$), however, the causal nature of the errors in BAYES indicated that a more extensive analysis of the differences in the two programs was warranted, and this is presented next.

4.5.2.2. Analysis of the Cases in which NESTOR and BAYES Differed

In 26 of the 50 cases evaluated, NESTOR and BAYES concluded different diagnoses. The expert showed a preference for one of the two programs' assessments in 24 of the 26 cases. Table 4-7 shows the number of cases rated superior for each program. NESTOR was superior in 13 cases (54% $\pm$ 10%) and BAYES in 11 cases (46% $\pm$ 10). This difference in performance is not statistically significant ($P = 0.74$).

	NESTOR	BAYES
Number of times rated superior	13	11

Table 4-7: The Comparative Ratings for the Cases in which NESTOR and BAYES Disagreed

In order to understand the source of the errors, a detailed analysis of the performance of each program on the 24 cases was undertaken.

4.5.2.2.1 Errors by NESTOR

Knowledge-base Errors. In 10 of the 11 cases in which BAYES' diagnosis was rated superior to NESTOR's, the cause was a deficiency in NESTOR's knowledge base (see Table 4-8. These were errors in which causal links were either missing or contained incorrect probabilities. For example, NESTOR believed that a serum calcium of 19 to 20 mg/100ml would always lead to a more severe depression of a patient's level of consciousness than mere lethargy, and this caused it to rate one case as clinically impossible (given the current disease set). BAYES succeeded in avoiding this diagnosis because it did not consider the interaction of the serum calcium level and the level of consciousness and therefore could not notice an inconsistency between them. However, according to the expert, lethargy *can* exist in this situation, so NESTOR was rated inferior. As an example of an error of complete omission, NESTOR did not know that pheochromocytoma can cause a resting tremor, although it knew that Graves' disease can. It therefore diagnosed Graves' disease in one case in which the expert considered an assessment of *insufficient data* to be preferable. BAYES fortuitously concluded *insufficient data* on the basis of an error in the way it aggregates evidence; this type of error will be discussed when the reason for BAYES' errors are discussed.

Using an Upper Bound as a Score. The remaining error made by NESTOR is more subtle. In this case two hypotheses were contending to be the most probable one: Graves' + pheochromocytoma vs. metastatic cancer + pheochromocytoma. The serum calcium was 14 to 15 mg/100ml. All three diseases can cause an increased serum calcium, thus in each of the two hypotheses the probability of a serum calcium was conditioned on the combined effects of two diseases. It happens that in NESTOR's current knowledge-base the bounds on the probability of Graves' disease causing a serum calcium of 14 to 15 is wider than the bounds of metastatic cancer causing it. Because of the way in which NESTOR aggregates evidence, this led to Graves' + pheochromocytoma having a higher upper bound probability than metastatic cancer + pheochromocytoma. The upper bound of the Graves' + pheochromocytoma hypothesis was not necessarily any more valid than that of the metastatic cancer + pheochromocytoma hypothesis, but it *was* more conservative. The problem arose because in diagnosing the cases for this evaluation, NESTOR used the upper bound of the score of an hypothesis as its scoring metric. Had

NESTOR required that the lower bound of the score of Graves' + pheochromocytoma be greater than the upper bound of the score of metastatic cancer + pheochromocytoma. it would have noticed that they overlapped and therefore would have concluded that this case contained insufficient data to determine a leading diagnosis. In this particular patient case, an assessment of *insufficient data* was preferred by the expert. BAYES was fortuitously able to conclude the preferred diagnosis of *insufficient data* because of an error in the way it aggregates data; this will be discussed next.

Type of Error	Number of Errors
Knowledge-base Deficiencies	10
Using Upper Bounds as a Score	1

Table 4-8: The Causes of NESTOR's Errors in the Cases in which NESTOR and BAYES Disagreed

4.5.2.2.2 Errors by BAYES

The reasons why BAYES was rated as inferior to NESTOR in 13 cases was analyzed and the results are summarized in Table 4-9. The three types of errors are each discussed below.

Type of Error	Number of Errors
Causal Inconsistencies	6
Causal Improbabilities	3
Improper Aggregation of Evidence	4

Table 4-9: The Causes of BAYES' Errors in the Cases in which NESTOR and BAYES Disagreed

Causal Inconsistencies. BAYES only represents the probability of each individual finding given a disease and in doing so it loses information about the causal consistency of a set of findings in the context of a given hypothesis. For example, BAYES allowed primary hyperparathyroidism to account for both stupor and a serum calcium level of 10.5 to 12 mg/100ml (among other findings), even though a serum calcium of such minor elevation is

incapable of causing stupor (see the left side of Figure 4-25). The problem is that primary hyperparathyroidism *can* cause stupor and it *can* cause a serum calcium of 10.5 to 12, but it can not by itself cause both at the same time. Some additional etiology is needed to account for stupor if the serum calcium level is only 10.5 to 12. In this particular case NESTOR recognized this and it introduced metastatic cancer as an additional disease in the hypothesis (see the right side of 4-25).

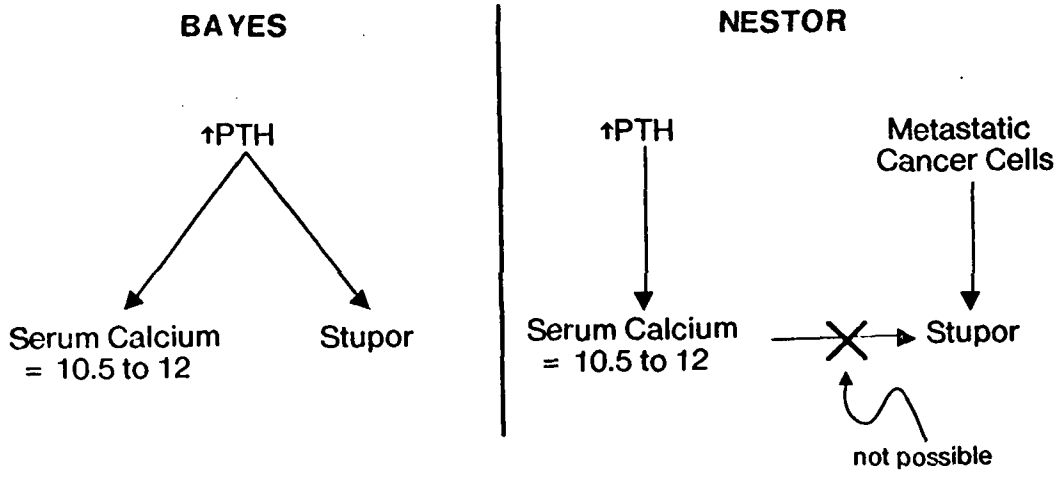


Figure 4-25: Causally Unstructured vs. Structured Findings:
Detecting Impossible Diagnoses

Causal Improbabilities. This is very similar to the causal inconsistency type of error just discussed. The difference is that although it was not categorically impossible for a given hypothesis to explain a given set of findings, nevertheless it was much less probable than the score that BAYES calculated by assuming the findings conditionally independent.

Improper Aggregation of Evidence. When BAYES scores a multiple disease hypothesis in which findings are caused by more than one of the hypothesis' diseases, it assumes that the diseases independently cause the findings. This may cause it to score the hypothesis too highly by ignoring causal dependencies among the findings.

For example, in one case sarcoidosis was able to account for all the findings including hypercalcemia, constipation, and lethargy. NESTOR knew that in sarcoidosis (SAR) the constipation (CONS) and lethargy (LETH) may be caused by the increased serum calcium (iCa) (see the bottom of Figure 4-26). NESTOR also considered the possibility that the best hypothesis was primary hyperparathyroidism (PHPT) + sarcoidosis. Again, both diseases may cause constipation and lethargy through the intermediate action of raising the serum calcium level. When primary hyperparathyroidism + sarcoidosis was scored by NESTOR, the decrease in the prior probability of this hypothesis (from that of sarcoidosis alone), was greater than the increased probability of the hypercalcemia given the two diseases (rather than hypercalcemia from sarcoidosis alone). In mathematical terms:

$$\frac{P(PHPT \ \& \ SAR)}{P(SAR)} \ll 1.0$$

and

$$\frac{P(CONS \ \& \ LETH \mid iCa) \times P(iCa \mid PHPT \ \& \ SAR)}{P(CONS \ \& \ LETH \mid iCa) \times P(iCa \mid SAR)} > 1.0$$

and therefore

$$\frac{SCORE(PHPT \ \& \ SAR)}{SCORE(SAR)} =$$

$$\frac{P(CONS \ \& \ LETH \mid iCa) \times P(iCa \mid PHPT \ \& \ SAR) \times P(PHPT \ \& \ SAR)}{P(CONS \ \& \ LETH \mid iCa) \times P(iCa \mid SAR) \times P(SAR)} < 1.0$$

where CONS = constipation, LETH = lethargy, iCa = increased serum calcium, PHPT = primary hyperparathyroidism, and SAR = sarcoidosis. Note that in the conditional probabilities above, the terms PHPT and SAR are used to represent the etiologies of primary hyperparathyroidism and sarcoidosis respectively.

Thus, NESTOR scored sarcoidosis as more probable than primary hyperparathyroidism + sarcoidosis.

On the other hand, BAYES did not realize the causal relationship between the serum

calcium level and the occurrence of constipation and lethargy (see the top of Figure 4-26). It therefore allowed both primary hyperparathyroidism and sarcoidosis to account independently for all three findings rather than just accounting for the hypercalcemia, which in turn would account for the constipation and lethargy. This led to the hypothesis primary hyperparathyroidism + lethargy having an increase in its ability to account for the findings (with respect to sarcoidosis alone) that was closely matched by the decrease in its prior probability (relative to sarcoidosis alone). In mathematical terms:

$$\frac{P(PHPT \ \& \ SAR)}{P(SAR)} << 1.0$$

and

$$\frac{P(CONS \ \& \ LETH \ \& \ iCa \mid PHPT \ \& \ SAR)}{P(CONS \ \& \ LETH \ \& \ iCa \mid SAR)} >> 1.0$$

and therefore

$$\frac{SCORE(PHPT \ \& \ SAR)}{SCORE(SAR)} = \frac{P(CONS \ \& \ LETH \ \& \ iCa \mid PHPT \ \& \ SAR) \times P(PHPT \ \& \ SAR)}{P(CONS \ \& \ LETH \ \& \ iCa \mid SAR) \times P(SAR)} = 1.0$$

The result was that the two hypotheses primary hyperparathyroidism + sarcoidosis and sarcoidosis were so close in their scores that BAYES incorrectly assessed this case as one in which there was insufficient data to select a leading diagnosis. In this case, the key difference between NESTOR and BAYES was that NESTOR causally structured the findings before scoring an hypothesis and BAYES did not.

In summary, all 13 of BAYES' errors were due to its inability to use causal knowledge. NESTOR had 10 knowledge-base errors and 1 error caused by ranking hypotheses according to the upper bound of their scores. The knowledge-base errors were corrected and all 50 cases were reevaluated by both NESTOR and BAYES. NESTOR's assessment of the

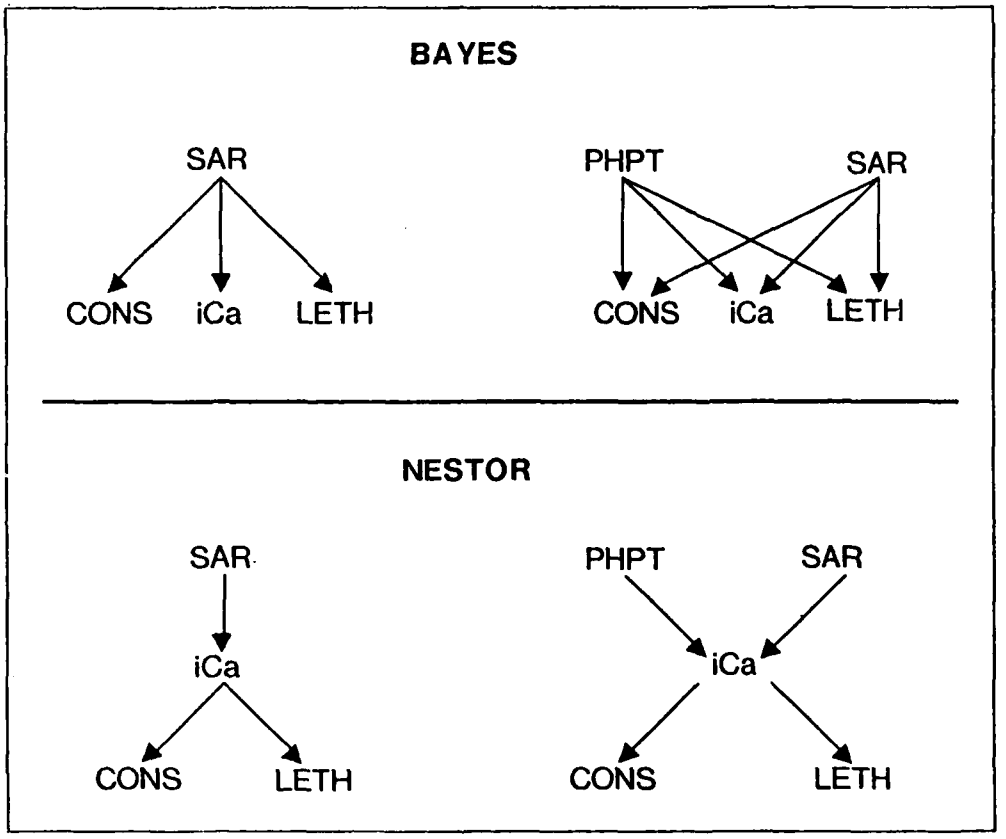


Figure 4-26: Causally Unstructured vs. Structured Findings:
The Effect on Aggregating Findings

10 cases with knowledge-base errors became the same as the original assessment by BAYES, which had originally received a superior rating by the expert. Also, the knowledge-base changes created no new cases in which NESTOR's assessment became inferior to either the original or the new assessment by BAYES. When the knowledge-base errors are eliminated from the comparison, NESTOR is rated superior in 13 of 14 cases (93% $\pm$ 7%) and BAYES in 1 of 14 cases (7% $\pm$ 7%), a statistically significant difference (two sided P = 0.0014).

4.5.3. Discussion

Overall both NESTOR and BAYES performed moderately well. However, several caveats are in order. First, the cases were not random samples taken from a real clinical population, but instead were generated from NESTOR's knowledge-base. Second, the expert who judged the cases in this study, also aided in constructing the knowledge-base. Third, the domain consisted of only 28 possible diagnoses (7 single disease and 21 double disease diagnostic possibilities). In general, the smaller the domain, the greater the diagnostic accuracy is expected to be, and this may have been a factor enhancing in the accuracy of the programs. Nevertheless, the study did uncover cases in which causal knowledge makes a critical difference in the accuracy of diagnosis as judged by an expert. Furthermore, some of the specific reasons why the causal knowledge is important have been elucidated.

The causal structuring of the interrelationships of the findings was found to account for the vast majority of the differences between NESTOR and BAYES, not counting correctable knowledge-base errors. The causal structure allowed NESTOR to recognize when the findings were either causally inconsistent with respect to a given hypothesis, or causally implausible enough such that another hypothesis was more likely (see Figure 4-25). It also allowed the proper aggregation of multidisease causal influences upon a finding so that evidence was accumulated toward the finding itself rather accumulating toward the finding and all its causal successors (see Figure 4-26).

Although causal structuring of the findings was clearly important, the use of NESTOR's knowledge-based techniques for calculating the local joint conditional probabilities (local JCP's) *within* a causal structure did not seem to make a difference in

NESTOR's diagnostic accuracy. As an example, refer to the top of Figure 4-26. The fact that NESTOR represented *increased serum calcium* (iCa) as an intermediary finding *was* important, but the particular technique it used to calculate the local joint conditional probability P(constipation & lethargy | increased serum calcium) was not critical. Recall that local JCP's are the probabilities of a set of findings (call them B) given their immediate causes (call them A). Assuming conditional independence of the effect at this *local* level means that the findings in B are considered to be conditionally independent given the findings in A. This assumption of independence would not have altered the diagnostic accuracy of NESTOR in the cases evaluated. There are at least two possible explanations for this. First, the frequency of cases in which *knowledge-based* local JCP calculations make an important difference may be very small --- so small that the current sample of 50 cases was unable to produce such a case. Second, the current domain size may be too small to generate cases in which there are sufficiently complex causal interactions among diseases to warrant the need for sophisticated local JCP calculation methods. Further research in a larger domain with larger case sample sizes is needed in order to investigate these questions.[44]

There was only one case out of fifty in which NESTOR received a rating of *incorrect* and this was due to its using the upper bound of hypothesis scores in order to rank order the hypotheses. This error was actually the product of decisions made in the *use* of NESTOR in this study, rather than an inherent flaw in NESTOR's design. Nevertheless, it was made because preliminary case runs showed that too often the bounds of scores overlapped, when in fact one hypothesis was clearly superior. This preliminary analysis was confirmed in the actual study where overlap in the scores of the top two hypotheses occurred in 31 of the 50 cases (62%). The bounds on the scores were not necessarily incorrect, but they were so large that the best hypothesis was buried in the partial ordering of all the hypotheses. There were two sources for the wideness of the bounds.

1. The probabilities in the knowledge-base are bounded. Since no particular probability distribution is assumed between the bounds, it is necessary to be

---

[44]This raises an important issue, namely, that the causal representation of diseases requires considerably more knowledge-acquisition effort than representations that lack it. It is difficult for a single person in the time course of a dissertation to represent even a small number of diseases, as NESTOR currently does. Therefore, the acceleration of future research progress in this area may require a group effort in order to causally represent a large number of medical diseases. This knowledge-base could then be used by individuals to experiment with particular diagnostic methods.

maximally conservative when combining bounds in the calculation of a score. This problem can be reduced if some knowledge is given to NESTOR about the distributions, but the typical lack of such knowledge was one of the initial reasons for simply using bounds.

2. The methods NESTOR uses to aggregate causal probabilities are fairly conservative, even though causal knowledge is used to reduce the bounds when possible. This source of wide bounds can be reduced in several ways. First, more knowledge can be made available with which to constrain the interpretation of probabilities using methods similar to those in Section 4.3.3.1.1. Second, more probabilities in the form of joint conditionals can be provided so that less knowledge-based aggregation of evidence is needed. Third, more radical assumptions can be made about how to aggregate probabilities; for example, if the use of independence assumptions in local joint conditional probability calculations is found to generally lead to satisfactory diagnostic accuracy then they could be used (see Section 4.3.3.1.2) rather than relying on detailed knowledge-based causal techniques as is currently done in NESTOR (see Section 4.3.3.1.1).

It is surprising that the use of upper bounds to rank order hypotheses led to only one diagnostic error. Using the upper bounds is actually a refinement of categorical reasoning. In categorical diagnostic reasoning the goal is to find all hypotheses that are not causally inconsistent with the findings, regardless of how improbable they may be. This is tantamount to using probabilities of only 1 and 0 in NESTOR, where a 1 is assigned to any probability which may be nonzero, and 0 is assigned otherwise. However, NESTOR goes one step further in using known probabilities in an effort to minimize the upper bound as much as possible. It does this by using prior probabilities, conditional probabilities between causally related findings, and causal knowledge. NESTOR can integrate such quantitative causal knowledge with qualitative, categorical knowledge by assigning the latter events or relationships a probability with a lower bound of 0.0 and an upper bound of 1.0.

A scoring method based primarily on the upper bounds of hypotheses would be well-suited to a ruleout diagnostic strategy. The upper bounds of weak hypotheses would be expected to be low. Those hypotheses that did not have low upper bounds, relative to the other hypotheses, would be considered as plausible candidates for further investigation (i.e., collecting additional findings). One criteria for concluding an hypothesis would be to

ruleout all but one hypothesis (i.e., all hypotheses but one would have an upper bound score of zero). Alternatively, when this is not possible, some hypothesis could be classified as a tentative working-diagnosis if the upper bound on its score was significantly greater than that of all other hypotheses.[45] At any rate, the initial success in this study with using the upper bound as a scoring metric for locating the most probable diagnosis suggests that it may have promising applications in more general medical domains. Therefore, further investigation of knowledge-based methods to accurately minimize the value of the upper bound probability of a diagnostic hypothesis seems warranted.

4.6. Extensions

A number of possible extensions to NESTOR's causal representation and scoring procedure are described below.

4.6.1. Caching Subtree Values

The scoring procedure outlined in Section 4.3 can be computationally very expensive. Suppose a given set of findings leads to a causal graph with N intermediate causal nodes and that each node has V possible values. The procedure in Section 4.3 iterates over all possible values for each node. This means that V^N causal graphs are scored. As an example, suppose V is 5 and N is 12. In this case $5^{12} = 2.4 \times 10^8$ graphs must be scored. This is a very large computational task and for all practical purposes it would probably take too long to complete. Some means of increasing efficiency is needed.

One way to increase the efficiency of the scoring process takes advantage of subtrees of the causal graph that can be computed just once and stored for repeated use. The procedure depends on the fact that a subtree of a causal graph interconnects to the rest of the graph through only one node.

As an example, Figure 4-27 shows a causal graph with the etiology as the root, the

---

[45] The proposed use of upper bounds does not of course preclude the use of lower bounds when they are useful.

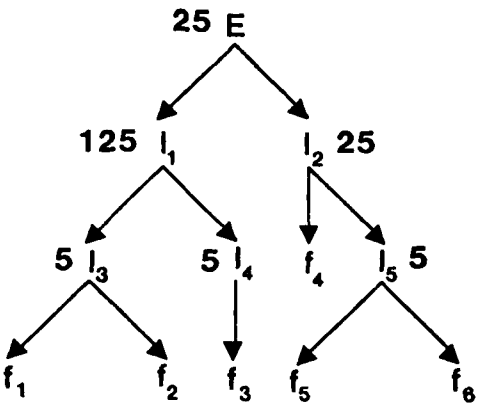


Figure 4-27: A Sample Causal Graph with Cached Values

findings as the leaves, and intermediate nodes connecting the etiology to the findings. The etiology and finding nodes will usually have only one value assigned to them. The intermediate nodes, however, may have several possible values. A brute force scoring process instantiates every possible value for the intermediate nodes, and computes the probability of each particular instantiation. The sum of the probabilities of all possible instantiations is P(F | E), the probability of the etiology causing the findings. If the five intermediate nodes in the example can each have 5 possible values, then the example requires that the probability of $5^5 = 3125$ instantiated graphs be determined. However, by caching subtree probabilities this number can be considerably reduced.

The caching process begins by computing the score of subtrees starting at the findings at the bottom and proceeding toward the etiology at the top. In Figure 4-27 the number of subtree instantiations is shown at the root of each subtree. For example the subtree of I_3 requires 5 instantiations since I_3 has 5 possible values and each of its sons has only 1

instantiated value since they are findings. For each value assigned to I_3 a probability is computed for its subtree, *given* that value of I_3. In this manner 5 probabilities are associated with the 5 values of I_3. Subtree I_1 is computed using the values of subtrees I_3 and I_4. Since each of these subtrees and I_1 itself has 5 possible values, the total number of instantiations to compute subtree I_1 is $5^3 = 125$. A similar procedure is used for the other subtrees. The total number of instantiations that must be considered is just the sum of the instantiations associated with each subtree, which is 190. This is a 16 fold savings over the 3125 required if a brute force technique is used.

In general the brute force technique requires $O(V^N)$ computation time, while the caching technique requires $O(N \times V^B)$, where B is the maximum branching factor in the graph.

Thus far the caching method has been presented as applicable only to a tree structured causal graph, however, a more general lattice structured graph may be cached. Although introducing non-tree-structured graphs complicates the evaluation somewhat, the caching procedure would still take advantage of possible computational savings.

4.6.2. Handling Unreliable Findings

Currently, NESTOR assumes the validity of all the findings it is given. In practice, there are false negatives and false positives among clinical data. NESTOR can be extended to handle this by treating the *actual state of a finding variable* as an intermediate variable, which will be termed f_{true}; intermediate variables generally have values that are never known with certainty. The *clinical information about a finding variable* is then treated as a finding is currently treated in NESTOR; this observable variable will be termed $f_{apparent}$. NESTOR's knowledge-base must then have probabilistic, causal links from the f_{true} to $f_{apparent}$, which in essence express the false positive and false negative values associated with $f_{apparent}$. These links can be treated like every other causal link in NESTOR. However, in terms of their semantics, the causal processes represented by these links are not limited to events of the patient, but also include those of the physician observer or the laboratory procedure being used.

4.6.3. General Multivariable Links

Currently, NESTOR's causal links have a single cause node connected to a single effect node. A useful extension would allow there to be multiple cause nodes and effect nodes. This would allow the expression of more complex probabilistic causal links.

Currently, a causal relationship is expressed in an explicit table that relates the probability of a given cause value leading to a give effect value. There is a need to extend the representation so that general mathematical relationships can be used to express the probabilistic relationships between cause and effect variables. The current domain does not require such a representation, since it contains few useful mathematical models. However, other domains such as acid-base disorders contain well known mathematical models that would need to be represented.

4.6.4. Arbitrary Intermixing of Categorical and Probabilistic Knowledge

The current representation places all the probabilistic knowledge at the top level of the causal model and all the categorical knowledge at the bottom level. A more general representation would allow an arbitrary number of levels and a mixture of both categorical and probabilistic knowledge within any level. This more complex representation will require more complex scoring procedures that are able to utilize all the available knowledge to narrow the bounds on the JCP calculations to a minimum.

4.6.5. Nosological Representation

In the current system nosological relationships are not represented. A nosological relationship defines a new node in terms of a logical combination (usually a disjunction) of existing nodes. An example of this is *increased serum level of immunoglobulins*. There are many different immunoglobulins, and a statement of this type allows the value of the whole class to be described rather than just the values of individuals of the the class. Pople [Pople82] discusses why integrating categorical causal and nosological models of disease can be beneficial in a diagnostic system. Future extensions of NESTOR could further advance this development toward more sophisticated representations by combining a categorical and

probabilistic causal representation with a nosological representation. The current scoring procedures would of course have to be extended to accommodate this more complex representation.

4.6.6. Generalizing the Temporal Representation of Links

Currently, NESTOR is only able to represent causes and effects that co-occur in time. A more general link representation is needed to express the temporal relationships in medicine in which the effect lags behind the cause. There has already been some research of temporal reasoning [Kahn 77, Fagan 80, Allen 81, McDermott 81, Mittal 82], but this is a broad and difficult area, and further investigation is needed. Chapter 3 discussed two reasons why a general approach to temporal reasoning is difficult in medicine: a lack of available temporal knowledge, and the computational complexity of the extra temporal dimension. Before offering some suggestions for handling these problems, the computational complexity issue will first be elaborated.

To make the analysis straightforward, consider that a link has only one cause variable called C and one effect variable called E, and that C and E are intermediate nodes with unknown values within some causal hypothesis being scored. Suppose that the there is some minimum granularity of time called T_g. No time measurement shorter than T_g is of interest. T_g might for example be a day. Furthermore, suppose that C and E have V possible values. The scoring procedure must consider every state of C and E. The state of E is some time pattern of its values from time t_{e1} to t_{e2}, discretized into time periods of length T_g. In the most general case, the values of C must be considered from time 0, the birth of the patient, until t_{e2}. In this case there are V^{t_{e2}/T_g} states of C to consider, and $V^{(t_{e2} - t_{e1})/T_g}$ states of E to consider. The total number of states to consider is found by taking the cartesian product of these two, and is $V^{(2 \times t_{e2} - t_{e1})/T_g}$. This general case can lead to an astronomical number of states to consider in scoring a diagnostic hypothesis. Suppose T_g is a day, and V is 5. If a 20 year old (7300 day old) patient presents with findings of 30 days duration, then this would lead to a causal simulation that must consider at least $5^{2 \times 7300 - 7270} = 10^{5123}$ states. Obviously, it is not possible to store this many probabilities or compute this many states. In fact the problem is worse than this, since we have only considered one link. This additional temporal dimension has escalated a computationally tractable problem to one that is

computationally untenable. To solve it, some major restrictions are needed to the general case just considered.

In the general case, C was presumed to have an influence on E from the birth of the patient to the end of E's duration. This is not very realistic. Usually a cause will have a more bounded span of influence. Let T_d be the maximum amount of time before t_{e2} that C can significantly influence E. In this case there are T_d/T_g possible time periods for the cause, and thus the cause may attain V^{T_d/T_g} states, while the effect variable may still attain $V^{(t_{e2}-t_{e1})/T_g}$ states. From this it is apparent that the number of states can be reduced in the following ways:

1. ***Decrease V***
   This amounts to considering coarser values of the variables. For example, instead of dividing the serum calcium level into intervals of 1 mg/100ml, divide it into intervals of 2 mg/100ml. Or, more drastically, just consider its values as being low, normal, and high. However, it is worth noting that coarser values in the representation of a variable, lead to less precision in the probability of the causal relationships in which that variable appears. The goal is to find the best balance between the tradeoffs of value granularity and probability precision.

2. ***Decrease T_d***
   The goal here is to minimize the length of the time period for which the cause can influence the effect. In many cases this may be quite small, as for example with drug effects. In other cases it may be long, as in the case of the serum calcium level causing renal stone formation.

3. ***Increase T_g***
   This may be one of the most effective means of decreasing the number of states to consider. Simulating the value of C and E for each day over the last year may be more detailed than is necessary, and perhaps a granularity of one month is sufficient. This increases T_g from 1 day to 30 days. Such an increase can drastically reduce the number of states considered. In fact, it may be useful to have T_g increase as a function of how far back in time the cause is referenced relative to the effect. In the last week the value of the cause on a daily basis may be important, but a year ago its value on a 3 month basis may suffice, in terms of validly calculating the probability of the effect value being considered. Note also that the T_g value for the cause variable need not be the same as the one for the effect variable.

Some combination of the above three methods will probably best reduce the amount of computation needed to score an hypothesis. Additionally, these methods will allow the expert to express causal relationships to a degree of temporal precision commensurate with clinical experience, rather than require probabilities be given for unrealistically precise or imprecise temporal patterns between causes and their effects.

4.6.7. Causal Feedback

So far, no examples have been given of causal feedback, because feedback implies a time delay between cause and effect, and time is not currently being represented in NESTOR. The extensions to the temporal representation just discussed, would allow feedback to be handled using the current scoring procedure. Essentially, the method would unfold a loop, using time to maintain a valid interpretation of the new graph. As an example, consider Figure 4-28. The causal graph on the left contains a feedback loop that is unfolded to form the graph on the right. The unfolded graph can be scored using NESTOR's current procedures. D_1 and D_2 represent the same node at different intervals of time. The only values of D_1 that are referenced are the ones that occur before the most recent value of B. Similarly, the only values of D_2 that are referenced are the ones that contain values of D that occur before the most recent value of E. The dotted line from D_1 to D_2 indicates that D_2 is not independent of D_1, since if D_1 and D_2 overlap in time.

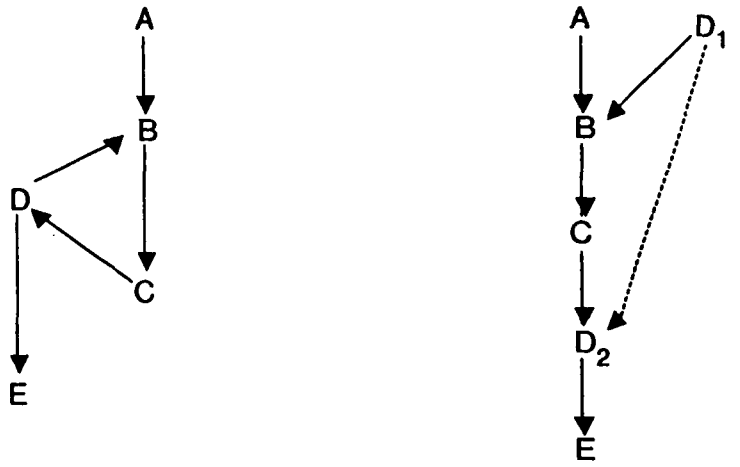


Figure 4-28: Unfolding a Feedback Loop

Chapter 5

Searching for the Most Probable Diagnostic Hypothesis

The goal of hypothesis generation and selection is to locate the most probable hypothesis that explains a given set of clinical findings. This task will be called hypothesis searching. Complex medical cases often involve two or more causally related diseases. A simple hypothesis search method that scores all single-disease hypotheses and returns the one with the highest score is inadequate in such situations. To deal with this problem, NESTOR allows an hypothesis to contain an arbitrary number of diseases from among those in its knowledge-base. The ability to consider multiple-disease hypotheses results in a very large number of diagnostic possibilities. For a domain with 100 diseases, there are $2^{100} > 10^{30}$ possible hypotheses. The enormity of this hypothesis space makes it clear that for general medical diagnosis some method is needed to reduce the number of hypotheses that must be considered. Intuitively, in most situations it seems unnecessary to consider *every* combination of diseases in order to find the combination that best accounts for a set of findings. This chapter deals with a method for formally making this intuition operational so that the most likely hypothesis can be determined without considering every possible hypothesis. The savings in the search time can often be substantial, and can transform a computationally intractable task into one that is readily completed.

5.1. Background

The key concepts in NESTOR's approach to hypothesis searching are drawn from the fields of operations research [Harvey 79, Hillier 74] and artificial intelligence [Winston 84]. The hypothesis search space for the diagnostic problem consists of all possible combinations of diseases. NESTOR's diagnostic search task involves searching this space for the most likely hypothesis that can explain the current set of findings. NESTOR uses a technique called best-first search with branch and bound pruning. The central idea of best-first search

is that hypotheses that are believed to be best are considered first. Thus, best-first search might be more clearly phrased as best-guesses-first search, since in general there is no guarantee that the hypotheses initially judged most likely are necessarily the best. Best-first search has at least two advantages. First, if there is insufficient time to locate the provably most probable hypothesis, the current best hypothesis can be reported. Second, locating highly probable hypotheses early in the search process allows the branch and bound pruning method to ignore entire areas of the search space.

The branch and bound pruning method is a means of disregarding provably non-optimal areas of the search space. As an introduction to this technique, consider the task of finding the shortest route from a home city H to a destination city D.[46] If a route from H to D of 10 miles is already known, then any route beginning at H and going through an intermediate city I that is more than 10 miles away from H can be ignored, since it is already worse than a known route of 10 miles. The savings results from not having to search for a route from I to D. This is a method of pruning (i.e., eliminating branches from) the search process.[47] In NESTOR a similar technique is used with diseases for cities and reciprocals of probabilities for distance, with the goal being to find a *pathway* of diseases (not necessarily a *single* disease) which explains all the findings with maximum probability (minimum distance). The pruning occurs when the extension of *an hypothesis* HYP (*routes from city H*) to include a particular *disease DX* (*city I*) can be proven to be always *less probable than* (*a longer distance than*) a currently known *hypothesis* (*route*). In this case the extension of the *hypothesis* HYP (*route from H*) containing *DX* (*city I*) can be pruned (ignored) because it is necessarily suboptimal. Section 5.3 will prove that this search technique is admissible, that is, it is guaranteed to find the most likely hypothesis.

---

[46] This example appeared in Section 1.5.2.2 of Chapter 1, but is repeated here for the convenience of the reader

[47] Note that the shorter the currently known *best* route, the more pruning that is possible. A known route from H to D of only 5 miles would generally allow more pruning than a route of 10 miles. This is why a best-first search strategy is desirable

5.2. The Precise Goal of NESTOR's Hypothesis Search Method

Chapter 4 demonstrates that the probability $P(F \& H_i)$, called the score of an hypothesis H_i given a set of findings F, can be used to rank order a set of hypotheses according to their likelihood. Thus, the goal of hypothesis generation is to find the hypothesis that maximizes $P(F \& H_i)$ over all possible hypotheses H_i, where H_i is some combination of diseases. Since NESTOR places lower and upper bounds on $P(F \& H_i)$, it may not be possible to determine a *single* most likely hypothesis if several of the most probable hypotheses have overlapping scores. The goal is therefore to find the hypothesis with the highest upper bound on its score.

There is no guarantee that the hypothesis with the highest upper bound will be the most probable hypothesis. However, NESTOR does use all its available knowledge and data to maximize the lower bound and minimize the upper bound on hypothesis scores in order to approach the true point probability of the hypothesis as much as possible. The evaluation of NESTOR's scoring methods in Chapter 4 demonstrated that in all but one of the 50 cases, the upper bound was sufficient to produce an acceptable top hypothesis.[48] Furthermore, the same techniques used to generate the top hypothesis (i.e., the one having the highest upper bound), can be used to generate all the hypotheses that have scores that overlap with the bounds of the top hypothesis. The most probable hypothesis will always be among this set.

5.3. An Example of NESTOR's Hypothesis Search Method

In order to convey the details of the search for the top hypothesis, consider a simple example in which the findings are anemia, weight loss, and a serum calcium level of 11.7 mg/100ml. The search proceeds as follows:

---

[48] The evaluation of the 50 cases in Chapter 4 used an exhaustive hypothesis generation method because the purpose of that study was to test NESTOR's scoring methods, not its hypothesis search methods. Section 5.7 in this chapter presents the results of an evaluation of the same 50 cases using a branch and bound search technique. The same diagnoses are produced (as expected since the search technique is admissible) with a significant reduction in the search time.

Restrict the disease set

The first step is to determine which diseases can causally influence the findings. This becomes the set from which all hypotheses are generated. In the present case all seven diseases currently in NESTOR's knowledge-base can have an influence since they all have a causal effect on the serum calcium level, which is one of the findings. The seven diseases along with their abbreviations and the upper bound of their prior probabilities are as follows:

MC: Metastatic cancer (other than myeloma)
Prior probability = 0.01
$\log_{10}$(Prior probability) = -2.0

G: Graves' disease
Prior probability = 0.01
$\log_{10}$(Prior probability) = -2.0

NMC: Non-metastatic cancer
Prior probability = 2×10^{-3}
$\log_{10}$(Prior probability) = -2.7

PHEO: Pheochromocytoma
Prior probability = 1.6×10^{-3}
$\log_{10}$(Prior probability) = -2.8

PHPT: Primary hyperparathyroidism
Prior probability = 10^{-3}
$\log_{10}$(Prior probability) = -3.0

SAR: Sarcoidosis
Prior probability = 2×10^{-4}
$\log_{10}$(Prior probability) = -3.7

MY: Myeloma
Prior probability = 10^{-4}
$\log_{10}$(Prior probability) = -4.0

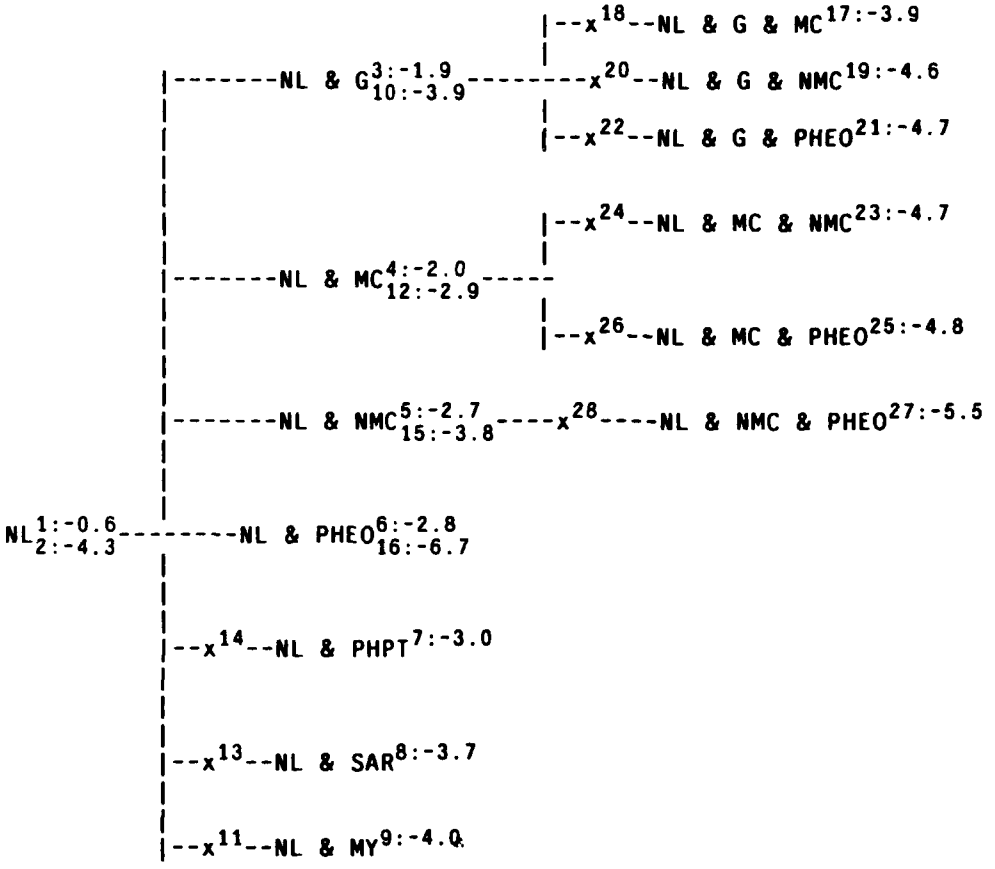
Guess a Diagnosis

Next, a heuristic hypothesis is generated to serve as the initial CURRENT.BEST.HYPOTHESIS. The score of the CURRENT.BEST.HYPOTHESIS will be used to prune non-optimal branches of the search tree. Any number of techniques can be used to generate the initial CURRENT.BEST.HYPOTHESIS, since it is not necessary that it be anything more than a guess (however, the better the guess, the more hypotheses that can be pruned). This is an area in which AI techniques for heuristically generating diagnostic hypotheses can be used to complement NESTOR's admissible search algorithm.

Several methods have been developed for generating an initial best-guess hypothesis in NESTOR. Section 5.7.1 discusses the one that is used most commonly. In the current example a less sophisticated method will be used; this will demonstrate how NESTOR finds the most likely hypothesis even when the initial guess is not a very good one. The simple method proceeds as follows: Suppose there is an unordered list of the diseases. NESTOR sequences down this list and if a disease can possibly account for a subset of the findings, then this disease becomes a member of the initial CURRENT.BEST.HYPOTHESIS, and this subset of findings is removed from the set of findings. This continues until all the findings have been accounted for. In the current example, suppose that myeloma (MY) is the first member of the list. Since myeloma can account for *all* the findings, it becomes the the initial CURRENT.BEST.HYPOTHESIS. Since any disease occurs in the context of some degree of normal physiological function, the CURRENT.BEST.HYPOTHESIS is more formally interpreted as a combination of myeloma and normal physiology (NL & MY). Next, NESTOR determines that the upper bound of the prior probability of NL & MY is $10^{-4.0}$ and its score is $10^{-5.8}$.

The next phase of the search process involves cycles of generating (i.e., branching), scoring (i.e., bounding), and pruning (i.e., eliminating) hypotheses. Figure 5-1 shows the result of these three processes in producing the entire search tree. The tree begins at the left with *normal physiology* (NL) as the only hypothesized state of the patient. The column to the right contains single disease hypotheses and the last column contains double disease hypotheses. Note that all hypotheses include NL, thus the pathophysiological mechanisms of diseases are always causally integrated with other normal physiological mechanisms.

The superscripts of the hypotheses in Figure 5-1 indicate (a) the stage in the search at which a node was generated and (b) the upper bound of the common log of its prior probability (logarithms are used for notational convenience). For example, normal physiology and Graves' disease (NL & G) was generated at step 3 in the search and the upper bound of its prior probability is $10^{-1.9}$; the log of the upper bound of the prior probability, such as -1.9 in this instance, will be called prior_{UB}. The subscripts indicate the stage at which the hypothesis was scored (if ever) and the score. For example, NL & G was scored at step 10 in the search and the upper bound of its score is $10^{-3.9}$; the log of the upper



LEGEND:

NL: Normal physiology	PHEO: Pheochromocytoma
G: Graves' disease	PHPT: Primary hyperparathyroidism
MC: Metastatic cancer	SAR: Sarcoidosis
NMC: Non-metastatic cancer	MY: Myeloma

Generic tree node: Hypothesis $^{\text{Step\#: } \log(\text{Prior}_{UB})}_{\text{Step\#: } \log(\text{Score}_{UB})}$

Example of a tree node: NL & $MC^{3:-1.9}_{10:-3.9}$

$x^{\text{Step\#}}$: Indicates a pruned branch at the given step number

Figure 5-1: An Example of an Hypothesis Tree Created During Hypothesis Search

bound of a score, such as -3.9 in this instance, will be called $score_{UB}$. An x on a pathway indicates that the hypothesis to the immediate right of the x will not be extended, since it is provably non-optimal. For example, x^{14} in Figure 5-1 indicates that the hypothesis NL &

PHPT and all its extensions are pruned at step 14. In order to explain how the tree was generated, the process will be divided into several phases.

Generate the Root of the Search Tree

As a first step, NL is generated as the root of the tree. It has a $prior_{UB}$ of -0.6, indicating that the prior probability that a patient will be physiologically normal is at most $10^{-0.6} = 25\%$. NL has a $score_{UB}$ of -4.3. This means that at most one in $10^{4.3}$ patients are expected to have anemia, weight loss, and a serum calcium of 10.5 to 12 mg/100ml without any underlying pathology as the cause. The anemia could be due to menstruation if the patient is a female, the weight loss due to diet, and the mildly elevated serum calcium due to a high normal calcium level.[49]

Generate, Score, and Prune Single Disease Hypotheses

Seven single disease hypotheses are generated in steps 3 to 9 according to the upper bound of their prior probabilities.

Next, each hypothesis is scored in turn. The hypotheses with the highest prior probabilities are scored first. Thus, NL & G is scored in step 10 and has a $score_{UB}$ of -3.9, because the upper bound of P(F & NL & G) is $10^{-3.9}$, where F is the current set of findings in the case. NL & G is then placed on a list called the HYPOTHESIS.LIST for later extension. The hypotheses on the HYPOTHESIS.LIST are ordered by the upper bound of their prior probabilities in descending order. Thus, NESTOR's best-first search strategy is to consider first those hypotheses that are most likely a priori. Since NL & G is the first hypothesis to be placed on the HYPOTHESIS.LIST, it is simply inserted there.

Recall that up to this point the CURRENT.BEST.HYPOTHESIS is NL & MY, which has a $score_{UB}$ of -5.8 and a $prior_{UB}$ of -4.0. Since the $score_{UB}$ of NL & G is greater than the $score_{UB}$ of NL & MY, NL & G becomes the new CURRENT.BEST.HYPOTHESIS, and the CURRENT.BEST.HYPOTHESIS $score_{UB}$ becomes -3.9. This immediately allows NL & MY to be pruned (see step 11 in Figure 5-1), since it has a $prior_{UB}$ that is less than -3.9.

---

[49]The normal range for lab values are commonly defined in terms of a 95% confidence interval for a physiologically normal population. Thus, 5% of the population is expected to fall outside this range and yet have no pathology.

To understand why NL & MY and all hypotheses that are extensions of it can never be the most probable hypothesis (given the current findings), consider the following general proof.

Suppose there is an hypothesis H such that P(H) < P(F & CURRENT.BEST.HYPOTHESIS), where F is the current set of findings. If H & * is some extension of H (that is, H & * is the combination of the diseases in H with some set of diseases different from those in H, denoted by *), then the following relations hold:

```
R1:     P(F | H & *) x P(H & *)   ≤   P(H & *)

R2.                    P(H & *)   ≤   P(H)

R3.                       P(H)    <   P(F & CURRENT.BEST.HYPOTHESIS)

Thus:   P(F | H & *) x P(H & *)   <   P(F & CURRENT.BEST.HYPOTHESIS)
 or equivalently:  P(F & H & *)   <   P(F & CURRENT.BEST.HYPOTHESIS)
```

Relation R1 is true since P(F | H & *) can be no greater than 1.0 according to the axioms of probability. R2 is true since the probability of a set of events (H & *) can be no greater than the probability of a subset of those events (H). R3 is a given relationship. Thus, the $score_{UB}$ of H & * is necessarily less than that of the CURRENT.BEST.HYPOTHESIS, and therefore hypothesis H and any extension of hypothesis H may be safely pruned. Notice that the ability to prune H & * depends on the $prior_{UB}$ of H being less than $score_{UB}$ of the CURRENT.BEST.HYPOTHESIS; in order to prune H & * it is *not* sufficient that the $score_{UB}$ of H be less than the score of the CURRENT.BEST.HYPOTHESIS, because some extension of H (i.e., H & *) may have a $score_{UB}$ that is greater than that of H, and more significantly, greater than that of the CURRENT.BEST.HYPOTHESIS.

For the current example, NL & MY is hypothesis H and therefore it and all possible extensions of it can be safely pruned from the search tree since its prior probability of -4 is less than the score of the CURRENT.BEST.HYPOTHESIS which is -3.9.

Next, at step 12, NL & MC is found to have a $score_{UB}$ of -2.9 and is placed on the HYPOTHESIS.LIST. Since -2.9 is greater than -3.9, NL & MC becomes the CURRENT.BEST.HYPOTHESIS and -2.9 its $score_{UB}$. This allows NL & SAR and NL & PHPT to be pruned in steps 13 and 14 according to the above proof.

In steps 15 and 16 NMC and PHEO are scored, but since neither of them has a $score_{UB}$ greater than the CURRENT.BEST.HYPOTHESIS, they are simply placed on the HYPOTHESIS.LIST. Note that they may *not* be pruned at this point, because although their $scores_{UB}$ are less than that of the CURRENT.BEST.HYPOTHESIS, their $priors_{UB}$ are not. Thus, it is possible that some extension of them will have a $score_{UB}$ that is greater than the CURRENT.BEST.HYPOTHESIS.

At this point all single disease hypotheses have either been scored or pruned.

Generate, Score, and Prune Double Disease Hypotheses

The HYPOTHESIS.LIST now contains (in order) the following four hypotheses: NL & G, NL & MC, NL & NMC, and NL & PHEO. The first hypothesis on the list, NL & G, is removed and its three extensions are generated. Note that PHPT, SAR, and MY are not considered as diseases in the extension of NL & G, since they were earlier pruned. Each of the three extensions of NL & G has a $prior_{UB}$[50] less than the CURRENT.BEST.HYPOTHESIS score of -2.9, so they are pruned. No new hypotheses have been added to the HYPOTHESIS.LIST.

In a similar manner NL & MC and NL & NMC are removed from the HYPOTHESIS.LIST, extended, and their extensions are pruned.

Finally, NL & PHEO is removed from the HYPOTHESIS.LIST, since there are no possibly successful extensions of NL & PHEO that have not already been tried.

At this point the HYPOTHESIS.LIST is empty, which means that all possible contenders for the best hypothesis have been explored. The CURRENT.BEST.HYPOTHESIS, metastatic cancer, is therefore the hypothesis with the highest upper bound that accounts for the current set of findings, and it is reported to the user. Altogether, 14 hypotheses and their prior probabilities were generated and 6 of these

---

[50]Currently, NESTOR computes the upper of the prior probability of an hypothesis as the product of the upper bounds of the prior probabilities of the individual diseases in the hypothesis. Section 4.3.3.2 discusses a method that does not assume that the disease etiologies are statistically independent of one another.

were scored.[51] Since there are $2^7 = 128$ possible hypotheses, a considerable savings in the amount of search has been achieved.

5.4. A General Branch and Bound Algorithm for Determining the Most Probable Diagnostic Hypothesis

A general branch and bound algorithm to locate the most probable diagnostic hypothesis is given in Figure 5-2. In the algorithm the phrase $prior_{UB}$ is again used as shorthand for *the upper bound of the prior probability.* Similarly, the word $score_{UB}$ is shorthand for *the upper bound of the score,* where scores are the values of P(F & H) as described in Chapter 4.

5.5. User Options to Direct Hypothesis Searching

There are several aspects of hypothesis formation that the user can explicitly control. Up to this point the default options have been used. The remainder of this section explains the other options available and how they are implemented.

5.5.1. Limiting the Amount of Search Time

The user can specify an upper limit on the amount of real-time that NESTOR should spend in searching for the most probable hypothesis. If this limit is reached, then NESTOR will display the most likely hypothesis found up to that point.

5.5.2. Finding the N Most Likely Hypotheses

Often it is useful to know some of the leading competitors to the top hypothesis. The user can explicitly inform NESTOR to generate any number of the top hypotheses.

Finding the top N hypotheses when N is greater than one is very similar to the method used when N = 1. Recall that the score of the CURRENT.BEST.HYPOTHESIS was used

---

[51] The initial heuristic guess plus the five shown in Figure 5-1 equals six.

- Determine which of the diseases among all the those in the knowledge-base can possibly have a causal influence on the current set of findings. Call this set DXS. This is done by using an inverted index in which every finding in the knowledge-base is associated with every disease that can be causally connected to the finding

- Use a heuristic method to compute an initial CURRENT.BEST.HYPOTHESIS. This step is merely an attempt to make a good guess at the most probable hypothesis; any number of techniques can be used. Compute the $score_{UB}$ of the CURRENT.BEST.HYPOTHESIS using the techniques in Chapter 4.

- The initial hypothesis is a one element set consisting of *normal physiology*. Place this on the list HYPOTHESIS.LIST. This list will always contain those hypotheses whose extensions are potentially better than the CURRENT.BEST.HYPOTHESIS. The list maintains the hypotheses in descending order of their $priors_{UB}$. Since hypotheses at the head of the list will be extended first, the order of the hypotheses in the list is a means of implementing a best-first search strategy. Other best-first strategies are possible by altering the ordering criteria of the HYPOTHESIS.LIST.

- While the HYPOTHESIS.LIST is not empty, cycle through the following steps:

  - Remove the first hypothesis from HYPOTHESIS.LIST and call it PARENT.HYPOTHESIS.

  - Generate the direct extensions of PARENT.HYPOTHESIS. A direct extension is an hypothesis that consists of the combination of the diseases[52] in PARENT.HYPOTHESIS with some disease in DXS that is not already in PARENT.HYPOTHESIS. All the hypotheses from this extension will be placed on a list called CHILDREN.HYPOTHESES.

  - For each CHILD.HYPOTHESIS in CHILDREN.HYPOTHESES, do the following:

    - If the CHILD.HYPOTHESIS has a $prior_{UB} \geq$ the $score_{UB}$ of the CURRENT.BEST.HYPOTHESIS then do the following:

      - Place CHILD.HYPOTHESIS in its appropriate location on the HYPOTHESIS.LIST.

      - Score the CHILD.HYPOTHESIS, that is, compute P(FINDINGS & CHILD.HYPOTHESIS) using the techniques in Chapter 4.

      - If the $score_{UB}$ of the CHILD.HYPOTHESIS is greater than the $score_{UB}$ of the CURRENT.BEST.HYPOTHESIS then do the following:

        - The CHILD.HYPOTHESIS becomes the CURRENT.BEST.HYPOTHESIS.

        - Remove (prune) the hypotheses in the HYPOTHESIS.LIST that have a $prior_{UB}$ less than the score of the CHILD.HYPOTHESIS.

- Report the CURRENT.BEST.HYPOTHESIS as the most probable hypothesis.[53]

Figure 5-2: A Branch and Bound Algorithm that Determines the Most Probable Diagnostic Hypothesis

---

[52] Although normal physiology is not actually a disease, it is treated as such by this step of the algorithm.

[53] When bounds are used, the CURRENT.BEST.HYPOTHESIS at this point is actually the one whose score has the highest upper bound

to prune the search tree. In the case of N > 1, there is instead a CURRENT.BEST.HYPOTHESES.LIST, which maintains the top N hypotheses discovered at any given point in the hypothesis generation process. The hypothesis on this list with the *lowest* score is used for pruning. This insures that no hypothesis is pruned that can possibly be more probable than any of the N top hypotheses.

5.5.3. Using Successive Sets of Assumptions to Find the Most Probable Hypothesis

The hypothesis generation procedure finds the hypothesis whose score has the highest upper bound. This hypothesis may have a score that overlaps with the score of other hypotheses. In other words, the *lower* bound of the top hypothesis (where the top hypothesis is the one with the highest upper bound) may be lower than the upper bound of some other hypothesis.

Recall that NESTOR is capable of scoring hypotheses using different sets of assumptions (see Section 4.3.3.1). These assumption sets are ordered, starting with those that are very conservative (i.e., likely to yield valid hypothesis scores) and ending with those that are relatively radical (i.e., those that are less likely to yield valid hypothesis scores). The most radical assumption set is always guaranteed to separate the bounds of the score of the top hypothesis from the score of all other hypotheses. Thus, if the user desires it, NESTOR can use successive sets of assumptions, starting with the most conservative, until it encounters a set that allows hypothesis generation to clearly distinguish a top hypothesis. The user is told the assumption set that was required in order to distinguish the top hypothesis; this is useful information for judging the validity of the result. The user can also alter the number and the order of application of the assumption sets so that hypothesis generation can be tailored even further.

5.5.4. Inclusion of User-Specified Diseases in All Hypotheses

NESTOR allows the user to specify a set of diseases that are believed to be present in the patient. The user may have access to data or knowledge not available to NESTOR, or the user may simply have an intuitive hunch that certain diseases are present. NESTOR will include these diseases in every hypothesis it generates. An hypothesis is a conjunction of diseases. If the user specifies the inclusion of M particular diseases in every hypothesis and some hypothesis is generated which contains N diseases, then M of the N diseases will correspond to those specified by the user, and the hypothesis will consist of the conjunctive occurrence of all N diseases.

This feature is implemented by augmenting the root of the initial search tree (normal physiology) with the user-supplied disease set (see Figure 5-1). Since every hypothesis in the tree is an extension of the root, all subsequently generated hypothesis will include these diseases. NESTOR also insures that the initial best-guess hypothesis contains the user-specified diseases as a subset.

5.5.5. Exclusion of User-Specified Diseases from All Hypotheses

The user may also instruct NESTOR to exclude a particular set of diseases from all hypotheses generated. These diseases are usually ones that the user has some reason to believe do not exist in the current patient. This feature is implemented by excluding the user-specified diseases from the initial disease set that NESTOR generates (see the first step in the algorithm in Figure 5-2).

The user may obtain the same result by explicitly specifying all the diseases to be used in hypothesis formation, rather than specifying those to exclude from the complete set of diseases. This is sometimes easier to do when the number of exclusions is large.

5.5.6. Limit the Number of Diseases in an Hypothesis

Sometimes it is cognitively helpful to first limit diagnosis to the consideration of hypotheses with a simple structure. For example, the user may want to first know if there is a single disease that can account for all the findings before having NESTOR explore more complex multiple disease hypotheses. Later, if a multiple disease hypothesis is found to be more probable, the user can have NESTOR compare it to the most probable single disease hypothesis in order to understand why the more complex hypothesis is preferable. NESTOR allows the user to specify that only hypotheses with N or fewer diseases should be generated in the search for the most probable diagnostic hypothesis.

This feature is implemented by restricting the depth of the search tree that NESTOR can explore. For example, if the user had set N to one, then in Figure 5-1 the last column consisting of double disease hypotheses would never have been generated.

5.5.7. A User-specified Goal for the Precision of the Posterior Probability of the Most Probable Hypothesis

The primary goal of the hypothesis search procedure is to determine the most probable hypothesis, given a set of findings. However, it is often useful to know the posterior probability of that hypothesis. For example, an hypothesis may be provably more probable than all the others, but if it has a probability of only 2%, then there would be little confidence in it. NESTOR is capable of *bounding* the posterior probability of an hypothesis, and the tightness of these bounds can be specified by the user. In order to do this, NESTOR must usually perform additional hypothesis generation after locating the provably most probable hypothesis. This allows it to calculate a tighter bounds on the probability of the findings, P(F). Recall that the posterior probability of the best hypothesis H, given a set of findings F, can be expressed as follows:

$$P(H \mid F) = \frac{P(F \,\&\, H)}{P(F)}$$

Tightening the bounds on P(F) results in tighter bounds on P(H | F), the desired posterior

probability. Section 6.2 explains the details by which both the lower and upper bounds of P(F) can be tightened by additional search.

5.6. Related Work

This section reviews the hypothesis generation methods of a number of diagnostic computer programs. The list of programs reviewed is not exhaustive, but instead is intended to discuss a representative sample of the types of techniques that are used for computer-based hypothesis generation. Table 5-1 outlines the major points of the comparison.[54] The features being compared are first outlined below. Then each system is discussed.

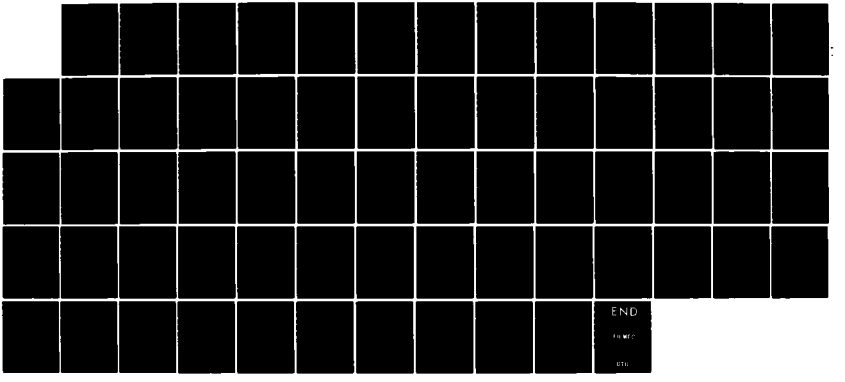
System	Disease Activation Technique	Type of Hypothesis Scored	Hypothesis Generation Technique	Form of the Final Diagnosis
BAYES	Exhaustive	Single dx	Activation	Single dx
MYCIN	Hierarchy	Dx class	Activation	Multiple dx
PIP	Triggers	Single dx	Activation	Multiple dx
INTERNIST	Triggers	Dx class	Sequential	Multiple dx
CASNET	Exhaustive	Single dx	Sequential	Multiple dx
ABEL	Triggers	Multiple dx	Exhaustive	Multiple dx
CADUCEUS	Findings	Multiple dx	Parsimony	Multiple dx
NESTOR	Findings	Multiple dx	Probability	Multiple dx

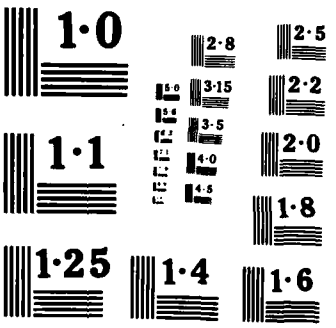
Table 5-1: A Comparison of the Hypothesis Search Methods of Selected Diagnostic Programs

[54] The comparison is meant to convey the *general* approaches of the programs, although methods for handling exceptions may exist.

AD-A152 046 NESTOR: A COMPUTER-BASED MEDICAL DIAGNOSTIC AID THAT INTEGRATES CAUSAL AN..(U) STANFORD UNIV CA DEPT OF COMPUTER SCIENCE G F COOPER NOV 84 STAN-CS-84-1031 3/3

UNCLASSIFIED N00014-81-K-0004 F/G 9/2 NL




1·0
1·1
1·25
1·4
1·6
1·8
2·0
2·2
2·5
2·8
3·15
3·5
4·0
4·5
5·0
MICROCOPY

5.6.1. A Description of the Features Being Compared

Disease activation technique

All the programs in Table 5-1 have some means of deciding which diseases are possible candidates for inclusion in the best diagnostic hypothesis. Some of the programs consider all diseases as candidates and therefore use an *exhaustive* technique. Others designate particular findings as *triggers* which, when they occur in a given case, are used to activate specific diseases. Other programs use all the *findings* as triggers. The *hierarchy* technique uses an hierarchy of diseases in order to determine which abstract disease classes to score. The leaf nodes of the hierarchy consist of individual diseases.

Type of Hypothesis Scored

Some programs can only score a *single* disease at a time, while others are able to score *multiple* diseases as a unified hypothesis. The programs that score disease *classes* are similar to those that score single disease hypothesis, except that an abstract class of diseases (e.g., cardiovascular disease) can also be scored as if it were a single disease hypothesis. In some programs that score a single disease (or disease class) at a time, the score of a disease may be influenced by the previous diseases that were concluded. However, the score of the concluded diseases will not change. In contrast, programs that score multiple disease hypotheses consider the mutual interactions of an entire set of diseases and score them as a whole.

Hypothesis Generation Technique

An *activation* technique of hypothesis generation uses the activated diseases as single disease hypotheses to be scored.

A *sequential* technique finds the single disease that best accounts for any subset of the findings. To do this it scores all the single diseases that have been activated. It then ignores the findings that are covered by this disease and repeats the process. This cycle continues until all findings are accounted for by some disease.

A *parsimony*-based method uses a heuristic technique in which the structurally simplest hypotheses are extended during hypothesis generation to find the simplest hypothesis that accounts for all of the findings.

An *exhaustive* technique scores all multiple disease hypotheses that can be formed by combinations of diseases that have been activated.

A *probability*-based method uses probabilistic knowledge about hypotheses, such as their prior probabilities, in generating and pruning hypotheses.

Form of the Final Diagnosis

This indicates whether the program can produce a final diagnostic assessment that consists of just a *single* disease or of *multiple* diseases.

5.6.2. Comments on Each Program

BAYES

The term BAYES is used to designate a class of programs that make specific assumptions in applying the general form of Bayes' formula. While these assumptions facilitate the *application* of the Bayes' formula, they also result in a system that is not as general, and sometimes not as accurate, as the direct application of the formula. In particular, BAYES is used to designate a large class of programs that use Bayes' formula under the assumption of conditional independence [Miller 77]. In addition, these programs typically have a knowledge-base that contains only a small number of diseases. They compute the posterior probability of each disease and report the one with the highest probability. Thus, they assume that the patient has only one disease.

MYCIN

MYCIN[55] [Shortliffe 76] uses rules to represent an implicit hierarchy (tree) of disease classes that are used during diagnosis. All the children of the activated disease classes (nodes) at depth i in the tree are first scored. The children that have a score greater than a preset threshold become the activated disease classes at depth i+1. This process of activation at successively deeper levels continues until some subset of the leaf nodes in the

---

[55]MYCIN is a general rule-based inference system that uses a goal-directed control structure [vanMelle 80]. The generality of its methods makes it difficult to discuss its diagnostic techniques in the abstract. Thus, the current discussion is based on the most recent instantiation of MYCIN that diagnoses meningitis [Yu 79a, Yu 79b].

tree is activated. These leaf nodes are individual diseases, and they form MYCIN's diagnostic hypothesis. Thus, MYCIN can conclude multiple disease hypotheses, but it has no way of generating and scoring multiple disease hypotheses as a unified pathological process.

PIP

PIP [Pauker 76, Szolovits 78] also scores only single disease hypotheses.[56] However, it does have knowledge of causal and associational links between diseases which it uses to influence the score of a single disease. The diseases to be scored are determined by a triggering mechanism that is sensitive to the current state of the diagnostic process: All diseases with triggers that match a finding or match a concluded disease are marked as activated and then scored. Also, any disease that is closely linked to an active disease, and that has any finding that occurs in the current case (it need not be a trigger), is made active and scored. PIP concludes the existence of a disease once its score surpasses a set threshold. Its relatively elaborate triggering procedure is an attempt to control the number of hypotheses that are activated and subsequently scored by using an heuristic hypothesis generation strategy.

INTERNIST

INTERNIST [Pople 75, Miller 82] has a set of findings associated with each disease which when present will trigger (activate) the disease. In fact, most findings serve as triggers. A higher level disease class within INTERNIST's disease hierarchy is activated if the diseases it subsumes in the hierarchy can all account for the same findings. The program uses a sequential hypothesis formation method. At each stage in hypothesis formation it scores all activated diseases and concludes the top disease if it has a score sufficiently greater than the next best contender.[57] Within a given stage, INTERNIST scores all activated diseases by assuming conditional independence of the findings. If a top disease can not be concluded, the user is asked additional questions about the patient's findings. Once a disease is concluded, the findings it accounts for are removed from the list of findings and, if there are

---

[56] Actually, an hypothesis can also be a clinical or a pathophysiological state.

[57] The top disease and another disease are contenders if the findings not explained by one of them are a subset of the findings not explained by the other

any remaining important findings, then the process recycles. A previously concluded disease can act like a finding in influencing the scoring of diseases in later scoring cycles. Thus, although INTERNIST does not score a unified set of diseases all at once, it does attempt to approximate this process. Eventually, when no findings of importance remain, a set of concluded diseases will constitute the diagnosis that is reported to the user.

This stepwise construction of a diagnosis in INTERNIST constrains the hypothesis search space to single disease hypotheses, but at the expense of never fully considering the possible pathophysiological interactions of a multidisease process.

CASNET

In the CASNET program [Weiss 78] findings are used to trigger pathophysiological states in a causal network. The states are then scored as either confirmed or denied using thresholds. These states are thereafter treated as though they were findings which must be explained. To do this, an exhaustive exploration of every etiology node in the causal graph is performed. An etiology node is any state in the causal graph that has no causal predecessors. The *best* etiology is the one that heads a pathway that includes the greatest number of confirmed nodes without including any denied nodes. This etiology is concluded to exist. The nodes accounted for by the concluded etiology are disregarded and the process recycles until all nodes are accounted for by some etiology. The set of generated etiologies constitutes the diagnostic hypothesis. The sequential nature of this process closely resembles that found in INTERNIST. Both programs only approximate the process of generating and scoring multidisease hypotheses as unified pathological conditions.

ABEL

ABEL [Patil 81] is a program that diagnoses acid-base disorders. In its initial hypothesis formulation it uses a special set of findings important in this domain (the serum electrolytes) to trigger and formulate all possible single and multiple disease hypotheses that are consistent with those findings. It then uses causal knowledge to score this hypothesis set with respect to *all* the patient findings. This allows it to prune those that are unlikely. ABEL is an improvement over the other programs discussed above because it is able to generate multiple disease hypotheses and score them as a unit, rather than in a stepwise fashion. This allows it to be sensitive to all the interactions among the diseases in the hypothesis. It is able

to use an exhaustive hypothesis generation strategy because the number of diseases in the acid-base domain is very small. However, such exhaustive enumeration of all multiple disease hypotheses is computationally intractable in more general domains of medicine where there are typically hundreds of diseases and therefore many billions of possible multiple disease hypotheses.

CADUCEUS

The CADUCEUS program has been designed by Pople as a successor of INTERNIST [Pople 82]. It is based on a knowledge representation which consists of a rich network of *causal* and *hierarchical* relationships among findings, intermediate causal states, etiologies, and disease classifications. Hypotheses can consist of either single or multiple disease states that are interconnected by these causal and hierarchical relationships. Findings serve to trigger particular nodes in the network. The goal of the hypothesis generation process is to find the simplest set of diseases that account for a given set of findings. Several syntactic criteria for measuring simplicity have been proposed, such as the number of unconnected subgraphs in an hypothesis. Using this criteria, the goal would be to find the hypothesis that has the minimum number of causally independent pathological processes. Although Pople notes the possibility of using prior probabilities and link weights to constrain hypothesis generation, to my knowledge no specific methods have been developed. Without some such constraining methods, the number of syntactically equivalent hypotheses may be too great to consider.

NESTOR

NESTOR, ABEL, and CADUCEUS all have the ability to generate and score multiple disease hypotheses, and in this sense they are more powerful than the other programs in Table 5-1. The key factor distinguishing NESTOR from all the other programs, including ABEL and CADUCEUS, is its ability to use the probabilities of hypotheses in constraining hypothesis generation. NESTOR is able to locate the most probable hypothesis without

resorting to exhaustive search or to reliance on heuristics such as Occam's razor.[58] Figure 5-3 shows how NESTOR's method for generating and selecting the most probable hypothesis relates to its other tasks. In particular, notice how the generation and selection task uses the hypothesis scoring task.

In surveying the literature for diagnostic programs that use techniques similar to NESTOR's, none where found. However, some research by Horning on the generation of formal grammars [Horning 69] was discovered which has some close parallels to NESTOR's hypothesis generation method. The goal of Horning's research was to develop a technique for finding the best grammar that could generate a given set of sentences. In an analogy to NESTOR, a grammar can be considered as a diagnostic hypothesis, and the sentences as the findings. By defining a criterion for calculating a prior probability for any grammar (P(G)) and calculating the probability of the sentences given a grammar (P(S | G)), he was able to use a branch and bound technique to constrain the generation of grammars in search of the one that best accounted for the sentences.

5.7. An Evaluation of NESTOR's Hypothesis Search Method

An formal evaluation of NESTOR's hypothesis generation technique was conducted in order to evaluate the method's effect on reducing the search required to locate the most probable hypothesis.

---

[58] Occam's razor is a rule of thumb which, in its popular form, states that simpler hypotheses are preferable to more complex ones. In the context of diagnosis it is generally interpreted as meaning that of all the muluple disease hypotheses that can possibly account for the patient's findings, the hypothesis with the fewest diseases is preferred. This method is not sensitive to the effect the prior probabilities of hypotheses and the conditional probabiliues of findings given hypothesis etiologies can have on the posterior probability of the hypotheses; it is possible that an hypothesis with N diseases is the most probable cause of a set of findings, even if some hypothesis with N-1 diseases can theoretically account for all of the findings.

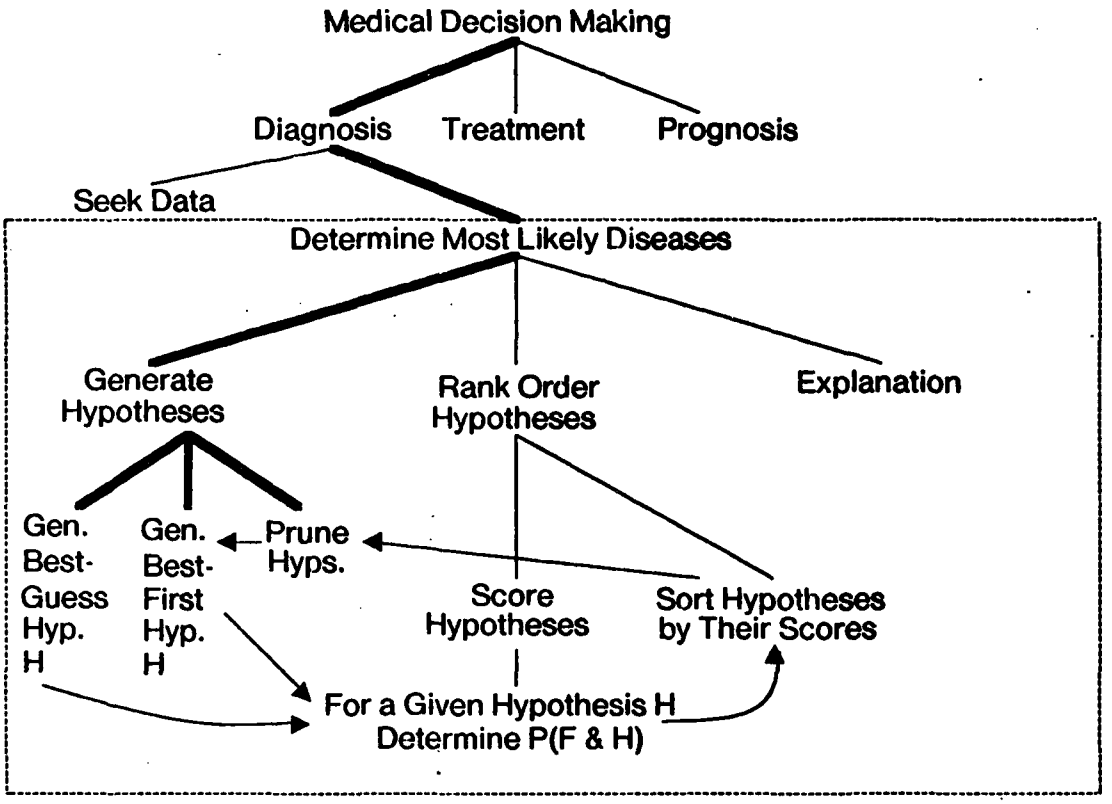


Figure 5-3: The Subtask of Searching for the Most Probable Hypothesis

5.7.1. Methods

A subset of the fifty cases used to evaluate NESTOR's scoring method (see Chapter 4.5.1.3) was used to evaluate its hypothesis generation method. Nine of the fifty cases were judged by NESTOR to be clinically impossible in the evaluation discussed in Chapter 4. These nine cases were eliminated since it is unlikely that inconsistent findings would be encountered this often in clinical practice and therefore using them would lead to undue bias toward excess hypothesis generation. Thus, forty one cases were evaluated.

The following are the three independent variables in the study:

1. **The Method for Generating the Initial Hypothesis**
   This is an heuristic method used to generate an initial guess hypothesis. There were two methods tested. The first heuristic technique, called H0, always generated normal physiology as the initial guess. The second, called H1, used a sequential diagnostic technique akin to that of INTERNIST. It sequenced through all the diseases in the knowledge-base one at a time in the order of the upper bound of their prior probabilities, from most to least probable. If a disease could possibly account for a subset of the findings (regardless of how improbably) it became a member of the initial hypothesis and the findings it accounted for were disregarded by later diseases in the sequence.

2. **The Best-first Search Strategy**
   The hypotheses that were pending expansion were ordered by two different methods. The first method, called B0, simply placed any newly generated hypotheses at the head of the list of the hypotheses to be expanded. This causes hypotheses to by generated in a depth-first manner; that is, a given hypothesis pathway is expanded until it is provably non-optimal. The second method, called B1, used the upper bound of the prior probabilities of hypotheses to order them.

3. **The Pruning Method**
   The first method, called P0, never pruned any hypotheses. This led it to generate all possible hypotheses. The second method, P1, used the upper bound of the score of the current-best-hypothesis to prune hypotheses based on the upper bound of their prior probability. This is the technique discussed previously in Section 5.4.

The following five different combinations of these three variables were tested:

1. **H0-B0-P0**: This is the baseline from which all improvements were judged, since it involves exhaustive search of the hypothesis space.[59]

2. **H0-B0-P1**: This will indicate the degree to which NESTOR's primary pruning method (P1) can decrease the search, even when used with a very simple initial guess method (H0) and best-first ordering strategy (B0).

3. **H1-B0-P1**: This will indicate how much search is saved by using a more sophisticated initial guess method.

4. **H0-B1-P1**: Similarly, this will test the amount of search saved by using a more sophisticated best-first ordering strategy.

5. **H1-B1-P1**: This combination tests the method that is currently used in NESTOR by default.

The above five combinations will be called search methods.

The following four dependent variables were measured:

1. **Hypotheses Generated**: This is the total number of hypotheses generated.

2. **Hypotheses Scored**: This is the total number of hypotheses scored.

3. **Order-of-best**: This is the order in the generation of hypotheses at which the most probable hypothesis was discovered. Note that in general it is necessary to do additional search after discovering the most probable hypothesis in order to *prove* that it is the most probable.

4. **CPU time**: This is the total computer execution time (in seconds) required to locate and prove the most probable hypothesis. This closely approximates the actual time a user would have to wait for NESTOR to generate and prove the best hypothesis if a DEC 2060 computer was solely dedicated to this one task.

---

[59] Combinations of H1 and B1 with P0 were not tested because P0 uses an *exhaustive* search technique and therefore using more sophisticated initial heuristic guesses (H1) or best-first ordering strategies (B1) could not possibly decrease the search time.

Finally, preliminary experiments indicated that the number of hypotheses generated was inversely proportional to the score of the best hypothesis. In order to investigate this relationship, a tally was made of the number of hypotheses generated as a function of the score of the best hypothesis when the search method H1-B1-P1 was used. Only the cases that had the best hypothesis generated initially by H1 were used, in order to control for the possible effect that a poor initial guess might have on the number of hypotheses generated.

5.7.2. Results

Table 5-2 shows the results for the five search methods applied to the forty-one cases. The format of the table can best be described by example. The second row of the table shows H0-B0-P1 as the search method tested. The first (top) number in any entry is the mean value of a variable. For example, H0-B0-P1 had a mean value of 17.8 seconds for the CPU time on the forty one cases. The standard deviation, shown in parentheses, was 21.6 seconds. The number below the mean value (91.3%) indicates that the mean CPU time was reduced by 91.3% in going from the 204.2 seconds of H0-B0-P0 (exhaustive search) to the 17.8 seconds of H0-B0-P1 (pruned search). The last three rows (i.e., H1-B0-P1, H0-B1-P1, and H1-B1-P1) in the table also provide comparisons with respect to H0-B0-P0. The P value (shown below the 91.3%) measures the statistical significance between the mean value of H0-B0-P0 and that of H0-B0-P1. In this case it is strongly significant ($P < 0.001$).[60]

Each of the four non-exhaustive search methods reached the same diagnosis as the exhaustive method H0-B0-P0 in all forty one cases. This is expected, since pruning technique P1 is admissible, that is, it guarantees that the best diagnosis will not be pruned.

Figure 5-4 is a plot of the relationship between the score of the best hypothesis and the number of hypotheses generated in the process of discovering this hypothesis and proving it to be the best one using the H1-B1-P1 search method. The best hypothesis is defined as the one with the highest upper bound on its score. The digits in the graph indicate the number of data points occurring at a given x-y coordinate. Only 21 cases appear in the graph because, as explained previously, only the cases that had the best hypothesis generated

---

[60] P values were computed using matched pairs analysis

Search Method	Hypotheses Generated	Hypotheses Scored	Order-of-Best	CPU time (seconds)
H0-B0-P0	128.0 (0.0)	80.2 (28.8)	25.9 (25.9)	204.2 (75.5)
H0-B0-P1	42.4 (28.4) 69.9% (p < 0.001)	8.0 (7.8) 90.0% (p < 0.001)	21.7 (20.2) 16.2% (p = 0.18)	17.8 (21.6) 91.3% (p < 0.001)
H1-B0-P1	36.9 (26.7) 71.2% (p < 0.001)	5.4 (4.9) 93.2% (p < 0.001)	9.5 (13.6) 63.3% (p < 0.001)	15.8 (16.1) 92.3% (p < 0.001)
H0-B1-P1	37.1 (25.4) 71.0% (p < 0.001)	6.4 (4.8) 92.0% (p < 0.001)	12.3 (6.0) 52.5% (p < 0.001)	13.9 (15.7) 93.2% (p < 0.001)
H1-B1-P1	36.0 (25.9) 71.9% (p < 0.001)	5.2 (4.8) 92.0% (p < 0.001)	6.9 (7.9) 73.6% (p < 0.001)	14.3 (14.4) 93.0% (p < 0.001)

Table 5-2: The Results of the Hypothesis Search Evaluation

initially by H1 were used, in order to control for the possible effect that a poor initial guess might have on the number of hypotheses generated.

5.7.3. Discussion

The results of each search method in Table 5-2 will be discussed in turn.

5.7.3.1. Exhaustive Search

The first row of Table 5-2 shows the results for the exhaustive search method (H0-B0-P0). The entire hypothesis space, consisting of every combination of the seven diseases, was searched in each case. Thus, $2^7 = 128$ hypotheses were generated for each case. On the average, only about 80 of the 128 hypotheses were scored because NESTOR does not score an hypothesis unless each finding in a given case can be causally influenced by at least one of the diseases in the hypothesis.

The best hypothesis was typically about the 26^{th} one generated; at first thought, this may seem surprisingly early to discover the best hypothesis since there are a total of 128 possible hypotheses. However, it occurred because the cases were originally generated from

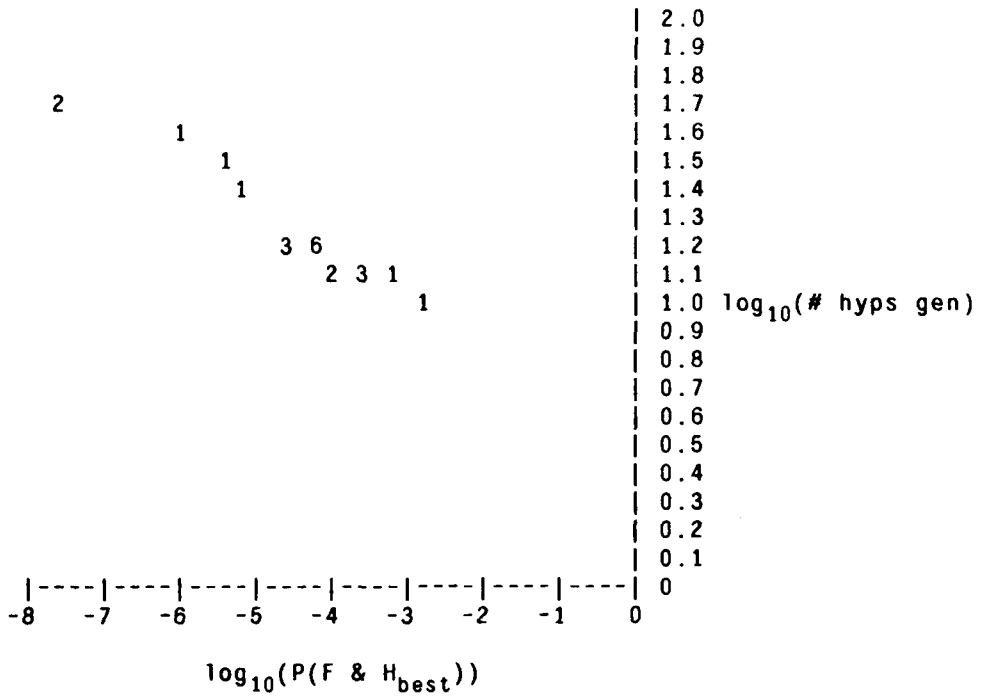


Figure 5-4: A Plot of the Relationship between the Number of Hypotheses Generated and the Score of the Best Hypothesis

one and two disease hypotheses. Thus, the hypothesis with the highest score was never very deep in the search tree and therefore was discovered relatively early in the search.

Each case required about three and a half minutes (204.2 seconds) on average to search all 128 hypotheses.

Although no test was made of H0-B1-P0, H1-B0-P0, or H1-B1-P0, these search methods would have also searched the entire hypothesis space, since no pruning would prevent it. This emphasizes an important point: the heuristic initialization and best-first search methods both serve to produce early pruning of non-optimal hypotheses, thus without a pruning technique, the rapid discovery of the best hypothesis will of course not reduce the search at all.

5.7.3.2. Pruned Search

The search method H0-B0-P1 pruned the search based on the prior probability of hypotheses. This led to a substantial reduction in the number of hypotheses generated and scored, which resulted in a 91.3% reduction in the search time. From this it is apparent that major savings in the amount of time required to locate the provably most probable hypothesis can be realized using the branch and bound pruning technique.

The order at which the best hypothesis was generated did not change significantly from that of the exhaustive search method ($p = 0.18$). This is reasonable, since H0-B0-P1 used the same unsophisticated methods as H0-B0-P0 to generate an initial guess hypothesis (H0) and to order the hypotheses being searched (B0).

5.7.3.3. Pruning with Heuristic Initialization and Best-first Ordering

The search method H1-B0-P1 again used P1 as a pruning technique, but in addition used H1 to generate an initial hypothesis. This caused a significant reduction in the order at which the best hypothesis was generated. However, when compared to H0-B0-P1 this led to only a minor decrease in the mean number of hypotheses generated and scored, as well as the average CPU time used per case. There are at least two reasons why this happened. First, the pruning technique by itself had already reduced the search to a marked degree, and therefore any further reduction would necessarily be relatively minor. Second, the H0-B0-P1 method was able to locate the best hypothesis at about the 22nd generation of an hypothesis. The primary reason for such early success was that the best diagnoses contained only one or two diseases. Thus, even without a sophisticated heuristic initialization method, it was able to locate the best hypothesis relatively early. This suggests that the heuristic initialization method may be most beneficial when there are complex cases involving three or more diseases. The small size of the domain that was tested may also be a factor here. In a larger domain there are a greater number of suboptimal pathways, and starting with a good initial guess could significantly reduce the number of them that are explored. Finally, the ability to locate a good guess early in the search has another advantage besides pruning: if the search process must terminate before completion, perhaps due to time constraints imposed by the user, then with heuristic initialization the best hypothesis at the termination is likely to be a good one, even if not the best.

The search method H0-B1-P1 used a more sophisticated best-first ordering algorithm than H0-B0-P1, but a crude heuristic initialization method. It achieved about the same results as the use of H1 in H1-B0-P1, and for the same reasons. It did not perform quite as well as H1-B0-P1 in finding the best hypothesis early in the search process. This is understandable since B1 must rely on the expansion of the search tree to locate the best hypothesis, whereas H1 can potentially guess it directly without any search.

The search method H1-B1-P1 combines the use of H1 and B1 in the context of pruning with P1. This is the default search method currently employed by NESTOR. The combination of H1 and B1 located the best hypothesis earlier in the search than either method alone (compare H1-B1-P1 to H1-B0-P1 and H0-B1-P1). This is probably due to H1 often suggesting a good hypothesis initially, while B1 is able to direct the search toward the best hypothesis even when H1 performs poorly. However, the improvement in locating the best hypothesis had a minimal effect on the number of hypotheses generated and searched. This is due to the same reasons that neither H1-B0-P1 nor H0-B1-P1 led to significant reductions over that achieved with H0-B0-P1. In fact, the CPU time for H1-B1-P1 is greater than that of H0-B1-P1. This suggests that H1 does not improve the search efficiency of B1 sufficiently to offset the cost of using H1. However, it is important to realize that the increase in search time over that of H0-B1-P1 is only slight and is not very statistically significant ($p > 0.1$). Also, the same issues that were raised above in the discussion of the utility of H1 in the context of H1-B0-P1 also arise here. Thus, if the domain increases in size and the best hypothesis becomes more complex, the combined use of H1 and B1 in locating the best hypothesis early in the search may significantly reduce the search time. Further testing will be required to know for certain.

To summarize the discussion of the results in Table 5-2, the use of a pruning technique was able to reduce markedly the search time required to locate and prove the most probable hypothesis. The addition of heuristic initialization and best-first search methods increased the rapidity of locating the best hypothesis significantly, but due to the small domain and the effectiveness of the pruning technique, this only led to a small decrease in the total search time.

In these experiments the use of a pruning technique achieved major savings. The

average search time was reduced by more than ten fold, from over three minutes to less than twenty seconds. Such improvement can mean the difference between user acceptance and rejection. The reductions in the experimental cases yielded a search time that seems quite acceptable. But, how general are these results? What will happened to the search time when the domain size, the number of findings, and the complexity of the most probable hypotheses increase? Will the efficiency gains of branch and bound pruning still be sufficient to yield acceptable search times? The next section discusses these issues.

5.7.3.4. The Extensibility of the Results

The graph in Figure 5-4 indicates that the number of hypotheses generated increases as the score of the best hypothesis, H_{best}, decreases.[61] This is due to the way in which the search is pruned. The smaller the score of H_{best}, the deeper the search tree must be explored before the prior probabilities of the hypotheses in the tree begin to be less than H_{best}. Recall that the P1 pruning technique does not prune the extension of an hypothesis until its *prior probability* is less than the score of the current best hypothesis, $H_{current.best}$. Since the score of $H_{current.best}$ can be no greater than that of H_{best}, this means that a small H_{best} score will cause more searching.

This relationship can be understood more quantitatively by considering the nature of the search tree. The tree consists of hypothesis nodes which contain diseases. At depth N in the tree are those hypotheses with N diseases. As a function of the tree depth, the number of diseases per hypothesis increases linearly. As a function of the number of diseases per hypothesis, the prior probabilities of the hypotheses decrease exponentially. Thus, as a function of the tree depth, the prior probabilities of the hypotheses decrease exponentially. Also, as a function of the tree depth, the number of hypotheses increases exponentially. Therefore, as the prior probabilities of hypotheses in the tree decrease exponentially, the number of hypotheses increases exponentially. Since for an hypothesis to be pruned its prior probability must be less than the score of the best possible hypothesis, as the score of the best possible hypothesis decreases exponentially, the number of hypotheses that must be

---

[61] Recall that H_{best} is the hypothesis with the highest posterior probability of all single and multiple disease hypotheses.

searched increases exponentially. This explains why the log-log plot in Figure 5-4 appears linear.[62]

The score of the best hypothesis is $P(F \& H_{best})$, where F is the set of findings in the case and H_{best} is the most probable hypothesis. As more findings are added to F the score will decrease in general, since the co-occurrence of N + 1 events can be no more probable than that of N events. In fact, the score will decrease exponentially as a function of the number of findings. To see this, consider the following:

$$1.\ P(F \& H_{best}) = P(F \mid H_{best}) \times P(H_{best})$$

$$2.\ P(F \mid H_{best}) = P(f_N \mid f_{N-1}, f_{N-2}, \ldots, f_1) \times P(f_{N-1} \mid f_{N-2}, f_{N-3}, \ldots, f_1) \times \ldots \times P(f_1 \mid H_{best})$$

In the great majority of cases, each term in the second equation is less than one, so multiplying N terms together will result in an exponential decrease in $P(F \mid H_{best})$ as a function of N. Thus, $P(F \mid H_{best}) < k_1^{|F|}$ where k_1 is a constant such that $0 \leq k_1 < 1$, and $|F|$ is the number of findings in F. Since $P(F \& H_{best}) = P(F \mid H_{best}) \times P(H_{best})$, the score of the best hypothesis also decreases as an exponential function of the number of findings in the case.

A similar phenomenon occurs as the number of diseases in H_{best} increases. Consider the following equation:

---

[62] Note that the plateau in the graph around 1.1 on the abscissa is due to the pruning of search after all single disease hypotheses had been considered.

3. $P(H_{best}) =$

$$P(D_{best_N} \mid D_{best_{N-1}}, D_{best_{N-2}}, \ldots, D_{best_1})$$
$$\times P(D_{best_{N-1}} \mid D_{best_{N-2}}, D_{best_{N-3}}, \ldots, D_{best_1})$$
$$\times \ldots$$
$$\times P(D_{best_1})$$

Each conditional probability is generally less than one, so that the prior probability of the best hypothesis decreases exponentially as a function of the number of diseases in the best hypothesis. Thus, $P(H_{best}) < k_2^{|H_{best}|}$, where k_2 is some constant such that $0 \leq k_2 < 1$, and $|H_{best}|$ is the number of diseases in H_{best}.

Combining the above equations yields the following result:

$$P(F \ \& \ H_{best}) < k_1^{|F|} \times k_2^{|H_{best}|}$$

This equation indicates that the score of the best hypothesis will decrease exponentially as a linear function of the number of findings in the case and the number of diseases in the best hypothesis. Recall that it has also been shown that the number of hypotheses generated will increase exponentially as the score of the best hypothesis decreases exponentially. Combining these two results leads to the conclusion that the number of hypotheses generated will increase exponentially as the number of findings in the case and the number of diseases in the most probable hypothesis increase linearly.

5.7.3.5. Summary

The performance of the pruning technique in this evaluation demonstrates that there exist cases in which the algorithm performs well. However, it was shown that even with pruning, the number of hypotheses generated will increase exponentially as the complexity of the case (i.e., the number of findings) and the complexity of the best hypothesis (i.e., the number of diseases in the most probable hypothesis) increases. The practical implications of this exponential character will require further study in a larger domain with a wider number of findings and diseases. However, it is important to realize that the derivation of the exponential relationship was based on using NESTOR's *current* admissible pruning technique in which prior probabilities are used as the pruning criteria. In the next section,

some suggestions are made for how to use other criteria in order to abate the exponential growth process.

5.8. Extensions

There are probably many ways in which NESTOR's current search technique can be made more efficient. Two potential methods for doing this are discussed next.

5.8.1. Improved Branch and Bound Pruning

The general goal of the hypothesis search method in NESTOR is to find the hypothesis H_{best} such that $P(F \& H_{best}) \geq P(F \& H_i)$ for all hypotheses H_i. In the ideal situation the exact maximum of $P(F \& H \& *)$ could be calculated for any set of findings F and any extension * of the hypothesis H. If this were possible then a search algorithm could determine at each depth of the tree which hypothesis to extend. If there were N diseases in the knowledge-base then the number of hypotheses scored at each extension would be on the order of N. If the best hypothesis contained M diseases then the search would need to extend to a depth of M in the search tree. Thus, on the order of N x M hypotheses would have to be scored. In essence the branching factor of the search would equal one, and little search would be needed to find H_{best}.

Also consider that as long as the score of the hypothesis being extended is not *underestimated*, it will not be incorrectly pruned. This is equivalent to saying that using the upper bound of the score of an hypothesis will never cause it to be improperly pruned. The ability to calculate the exact upper bound of the score of any extension of an hypothesis was just shown to lead to a branching factor of one. Unfortunately, this ideal situation is seldom achievable. Usually the upper bound is not so tight. This leads to a branching factor greater than one because at any given depth in the search tree there are typically many hypotheses that have extensions that have upper bounds on their scores that are greater than the score of the current best hypothesis. Thus, they can not be pruned. The more the upper bounds can be minimized, the more the search tree can be pruned, and the less time it will take to insure that the best hypothesis has been located.

NESTOR currently uses the upper bound on P(H), of hypothesis H which is being extended, as the upper bound on the score of H and any extension of H (represented as H & *). The following relationships show why this is so:

$$P(F \,\&\, H \,\&\, *) \;=\; P(F \mid H \,\&\, *) \times P(H \,\&\, *) \;\leq\; P(H \,\&\, *) \;\leq\; P(H)$$

Thus, if P(H) is less than the score of the current best hypothesis, the score of H and any of its extensions must also be less. The problem is that P(H) is not a very tight upper bound on P(F & H & *). If a tighter upper bound could be calculated it would lead to a smaller branching factor and therefore to less search.

One possible way in which this can be accomplished is suggested by the following relationships:

$$\begin{aligned} P(F \,\&\, H \,\&\, *) \;&=\; P(F \mid H \,\&\, *) \times P(H \,\&\, *) \\ &\leq\; UB[\, P(F \mid H \,\&\, *)\,] \times P(H \,\&\, *) \\ &\leq\; UB[\, P(F \mid H \,\&\, *)\,] \times P(H) \end{aligned}$$

UB is used to designate an upper bound. The key is in the calculation of UB[P(F | H & *)]; in NESTOR this term was set to one, the trivial upper bound. Any method to minimize this upper bound would result in reduced searching. One technique for doing this involves finding a subset of the findings in F, call them G, which can not be causally affected by the etiologies of any possible extension of H. In this situation the following relationship would exist:

$$P(F \mid H \,\&\, *) \;\leq\; P(G \mid H)$$

Thus, P(G | H) serves as a nontrivial upper bound of P(F | H & *).

There are perhaps many other ways of minimizing the upper bound of P(F & H & *); this is an area in which additional research is needed.

5.8.2. Searching in an Abstraction Space

Another technique to reduce search is to decrease the number of diseases in the hypothesis space. This is possible if the diseases can be classified into some hierarchy; this allows more abstract classes of diseases to be searched. For example, in the current domain the causes of hypercalcemia could be grouped into the abstract classes of *endocrine*, *cancer*, and *other*. The search could use these three as the "diseases" from which hypotheses are constructed. The total number of possible hypotheses is thus reduced from the current $2^7 = 128$ to $2^3 = 8$. This raises the important issue of the appropriate level at which to formulate a rank-ordered differential diagnosis. Factors that influence the appropriate level include not only the computational search time, but also the usefulness of the final diagnosis with regard to guiding medical diagnostic and therapeutic action. These issues need to be investigated.

This suggests the possibility of combining branch and bound pruning methods with a representation like that proposed for CADUCEUS by Pople [Pople 82]. The representation would contain a network of causally linked states which are grouped into hierarchies. The causal links would have associated conditional probabilities[63] at each level in the hierarchy. Nodes in the network would have prior probabilities.[64] Then, possibly some form of a branch and bound technique could be used to hierarchically search this network for the most probable diagnostic hypothesis to account for a set of findings. This is a largely undeveloped area of medical decision making research and appears to have significant promise.

---

[63] Actually, there would be *bounds* on the conditional probabilities.

[64] More precisely, there would be *bounds* on the prior probabilities.

Chapter 6

Explanation

The ability to explain a diagnosis is an important component of any medical diagnostic computer program, especially one that is designed to directly interact with a physician. Within the health care system it is the physician who has the ultimate medical, ethical, and legal responsibility for the care of the patient. Accordingly, physicians want and need to know the justifications for their diagnostic and therapeutic actions. A computer program that explains its diagnostic reasoning provides a physician with a basis for accepting or rejecting its advice in light of the physician's knowledge of the patient, knowledge of medicine, and common sense knowledge.

NESTOR has two explanation commands: COMPARE and CRITIQUE. The COMPARE command instructs NESTOR to contrast how well two diagnostic hypotheses account for the current set of clinical findings. Causal knowledge is used both in generating the explanation and in scoring the two hypotheses. The hypotheses being compared can be either a user hypothesis vs. a user hypothesis, a computer-generated hypothesis vs. a computer-generated hypothesis, or a user hypothesis vs. a computer-generated hypothesis. Thus, users can formulate a number of their own hypotheses and have them compared to computer hypotheses which have been generated under possibly differing user-specified constraints (see Section 5.5). In this way NESTOR provides a flexible diagnostic tool for the user to experiment with many ideas about the etiology of a patient's problem.

The critiquing command is similar to the COMPARE command, except that a single hypothesis is critiqued abstractly with respect to *all* other diagnostic possibilities rather than just one other hypothesis. The hypothesis being critiqued can be either user-generated or computer-generated.

Figure 6-1 shows how NESTOR's explanation commands relate to other medical decision-making tasks. This chapter discusses how these two commands are implemented.

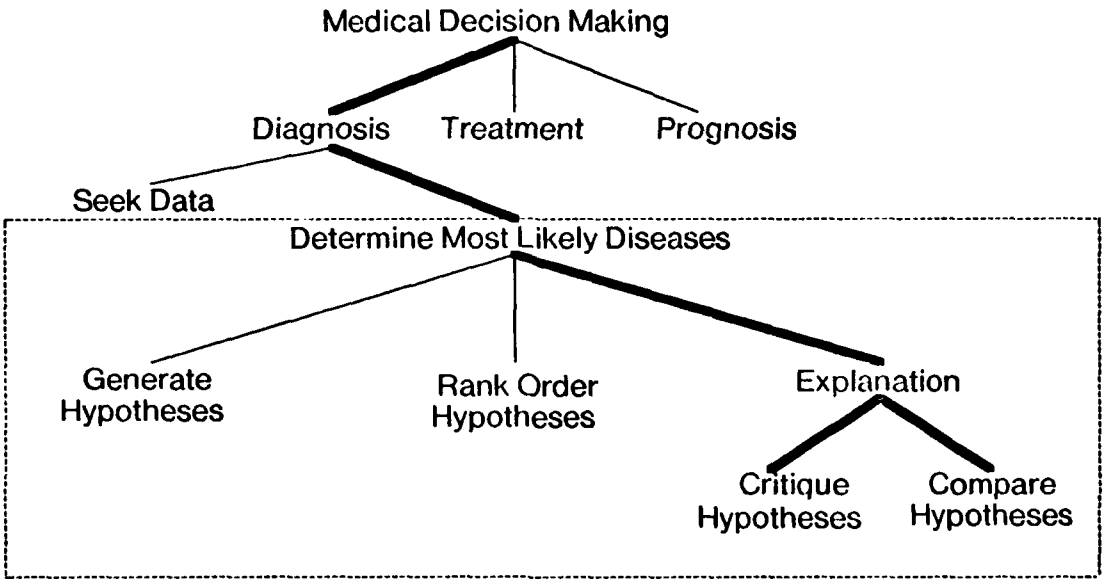


Figure 6-1: The Subtasks of Critiquing and Comparing Diagnostic Hypotheses

6.1. The COMPARE Command

The COMPARE command consists of two stages. First, for each of the two hypotheses being compared, NESTOR prints an English text version of the causal relationships linking the proposed etiologies of the diseases in the hypotheses to the findings in the current case. This case-specific causal knowledge gives the user a context in which to interpret the next stage of the comparison.

The second stage of a comparison informs the user how each of the current findings influences the relative probabilities of the two hypotheses. This complements the previous output of the qualitative causal structure of the hypotheses by giving the user a *quantitative, probabilistic* sense of their causal differences. Figure 6-2 shows an example of the COMPARE command that originally appeared in Chapter 2. In this example, even though an increased serum calcium can cause a decrease in a patient's level of consciousness, an increased serum calcium of 13.5 mg/100ml does not cause coma (i.e., the probability equals zero), because 13.5 is not high enough. Currently, NESTOR is not able to explain this type of causal detail, however, Section 6.4 proposes a method by which the current representation can be used to do this. The methods used to implement the two stages of the comparison command are discussed next.

6.1.1. Step 1: Text Generation for Explaining Case-Specific Qualitative Causal Knowledge

NESTOR creates an internal knowledge structure that represents the causal links between the etiology of a disease and the findings in a given case.[65] This acyclic, directed, causal graph is used in calculating the score of an hypothesis and also in explaining it. To explain the qualitative causal interactions, NESTOR performs a direct translation of the graph into English.

The first step in producing a translation is to order the set of findings. NESTOR begins by separating the non-causal findings such as age and sex from the causal ones such as muscle weakness. Here the term *causal finding* is being used to mean those findings that are causally influenced by the etiologies of the diseases of the hypothesis being explained. Next, the causal findings are partially ordered by their causal precedence. If finding f_1 causes finding f_2, then f_1 occurs before f_2 in the ordering. If f_0 also causes f_2, then both f_1 and f_0 occur before f_2 in the ordering. In addition, the ordering is constructed in a breadth-first manner consistent of course with the previous ordering rule. Since the two hypotheses

---

[65] See Chapter 3 for a discussion of the representation of case-specific causal graphs and the assumptions made about the temporal relationships among the findings. See Section 4.3.1 for a discussion of the construction of a case-specific causal graph.

```
* DISPLAY FINDINGS
1) SEX = FEMALE
2) TOTAL SERUM CALCIUM = 13.5 MG/100ML
3) LEVEL OF CONSCIOUSNESS = COMA

* COMPARE USER-HYP1 NESTOR-HYP1

-------------------------------------------------------------------------
THE FOLLOWING CAUSAL MODEL EXPLAINS THE MECHANISMS WHEREBY THE
USER-HYP1 HYPOTHESIS CAUSES THE FINDINGS:

 PRIMARY HYPERPARATHYROIDISM IS CHARACTERIZED BY THE PRESENCE OF INCREASED
 PTH. INCREASED PTH IS CAPABLE OF CAUSING AN INCREASED TOTAL SERUM CALCIUM.
 THIS IN TURN IS ABLE TO CAUSE A DECREASED LEVEL OF CONSCIOUSNESS.

-------------------------------------------------------------------------
THE FOLLOWING CAUSAL MODEL EXPLAINS THE MECHANISMS WHEREBY THE
NESTOR-HYP1 HYPOTHESIS CAUSES THE FINDINGS:

 METASTATIC CANCER IS CHARACTERIZED BY THE PRESENCE OF METASTATIC CANCER
 CELLS. METASTATIC CANCER CELLS CAN RESULT IN AN INCREASED TOTAL SERUM
 CALCIUM, AND TUMOR FORMATION IN THE BRAIN. TUMOR FORMATION IN THE BRAIN
 AND INCREASED TOTAL SERUM CALCIUM ARE CAPABLE OF PRODUCING A DECREASED
 LEVEL OF CONSCIOUSNESS.
-------------------------------------------------------------------------

THE TABLE BELOW CONTAINS THE PROBABILITY OF THE USER-HYP1 HYPOTHESIS
UNDER THE ASSUMPTION THAT IT AND THE NESTOR-HYP1 HYPOTHESIS ARE THE
ONLY HYPOTHESES POSSIBLE.
NOTE: UNDER THIS ASSUMPTION THE PROBABILITY OF THE NESTOR-HYP1 HYPOTHESIS
IS JUST 100 MINUS THE PROBABILITY OF THE USER-HYP1 HYPOTHESIS.

                                    P(USER-HYP1 | F1 TO Fn)

          FINDING                 NUMERIC FORM      GRAPHIC FORM
                                    (PERCENT)         (PERCENT)
                                                  0              100
-------------------------------------------------------------------------
F1. NO FINDINGS AVAILABLE         |   4 TO 17    |****              |
F2. SEX IS FEMALE                 |   6 TO 21    | ****             |
F3. TOTAL SERUM CALCIUM IS 13.5   |   5 TO 47    | *********        |
F4. LEVEL OF CONSCIOUSNESS IS COMA|      0       |*                 |
-------------------------------------------------------------------------
```

Figure 6-2: An Example of the Comparison of Two Hypotheses

being compared may have a different causal structure, the structure of the first hypothesis is used by default to produce the ordering.

The partially ordered list of findings directs the sequence in which the causal links are printed. The first nodes on the list are always the etiologies of the hypothesis, and a single sentence informs the user about them. Other nodes may be either findings or intermediate nodes that causally link findings to each other. A node is first removed from the list and all the case-specific causal links for which it is a cause are located. These causal links will have the following format:

```
(<relationship> <cause variable> <effect variable>)
```

The relationship describes whether the effect variable is a monotonically increasing or decreasing function of the cause variable.[66] The following is an example of the internal representation, along with its translation:

```
Internal representation:

   (<decreasing fn> <serum calcium> <level of consciousness>)

Findings:

  1. Serum calcium increased
  2. Coma (i.e., a decreased level of consciousness)

Translation to English:

   INCREASED SERUM CALCIUM CAN CAUSE A DECREASED LEVEL OF
   CONSCIOUSNESS.
```

There are several stylistic refinements that make the output text less mechanical and more readable. The phrase *can cause* in the above example is a linking phrase, and NESTOR uses several other linking phrases which are synonymous to it. Some examples are: *can lead to*, *can produce*, and *is capable of causing*. The phrases are used randomly, except that a phrase may not be used consecutively. Also, if a causal node has several effects which have the same relationship to it, then these effects are placed together as a compound

---

[66] NESTOR also has the ability to represent the relationship as unknown or as neither strictly increasing nor decreasing, but these are seldom used in the current domain.

phrase at the end of the sentence rather than in separate sentences. Similarly, if an effect node has several causes which have the same relationship to it, then these causes are placed together as a compound phrase at the beginning of the sentence. Finally, if the effect node of sentence$_i$ is also the cause node of sentence$_{i+1}$, then the cause node of sentence$_{i+1}$ may use a phrase such as *This in turn* in order to sound more fluent. Figure 6-3 shows the use of several of these refinements in explaining the more detailed links of causal model of the first hypothesis in Figure 6-2.

```
THE FOLLOWING CAUSAL MODEL EXPLAINS THE MECHANISMS WHEREBY THE
USER1 HYPOTHESIS CAUSES THE FINDINGS:

PRIMARY HYPERPARATHYROIDISM IS CHARACTERIZED BY THE PRESENCE OF INCREASED
PTH. INCREASED PTH CAN CAUSE AN INCREASED RENAL RESORPTION OF CALCIUM,
AN INCREASED GASTROINTESTINAL ABSORPTION OF CALCIUM, AND AN INCREASED
OSTEOCLAST ACTIVITY. INCREASED OSTEOCLAST ACTIVITY CAN RESULT IN AN INCREASED
RESORPTION OF BONE. INCREASED RENAL RESORPTION OF CALCIUM AND INCREASED
GASTROINTESTINAL ABSORPTION OF CALCIUM AND INCREASED RESORPTION OF BONE
ARE CAPABLE OF PRODUCING AN INCREASED TOTAL SERUM CALCIUM. INCREASED TOTAL
SERUM CALCIUM CAN CAUSE A DECREASED NEURONAL AND NEUROMUSCULAR FUNCTION.
DECREASED NEURONAL AND NEUROMUSCULAR FUNCTION CAN PRODUCE A DECREASED
LEVEL OF CONSCIOUSNESS.
```

Figure 6-3: Using Detailed Causal Knowledge in an Explanation

6.1.2. Step 2: Comparing the Quantitative Likelihood of Two Hypotheses

The second stage of the comparison involves a probabilistic assessment of how well the two hypotheses account for the findings. Recall that NESTOR assigns $P(F \& H_i)$ as the score of hypothesis H_i with respect to findings in F. The ratio of the scores of the hypotheses indicates the ratio of their posterior probabilities:

$$\frac{P(H_1 \mid F)}{P(H_2 \mid F)} = \frac{P(F \& H_1) / P(F)}{P(F \& H_2) / P(F)} = \frac{P(F \& H_1)}{P(F \& H_2)}$$

It is the information in this ratio that NESTOR conveys to the user. However, notice that as $P(F \& H_2)$ approaches zero the ratio approaches infinity. In order to have a finite scale for graphical output, NESTOR uses the function in Figure 6-4 to convey the same information.

$$P^*(H_1 \mid F) = \frac{P(F \& H_1)}{P(F \& H_1) + P(F \& H_2)}$$

Figure 6-4: The Statistic Used to Compare Two Hypotheses

The $P^*(H_1 \mid F)$ function indicates that this is the posterior probability of H_1 under the assumption that it and H_2 are the only two possible hypotheses. Thus, this statistic has a satisfactory intuitive interpretation and also has a finite value range from zero to one. It conveys the same information as the ratio of the posterior probabilities, to which it has the following relationship:

$$\frac{P^*(H_1 \mid F)}{1 - P^*(H_1 \mid F)} = \frac{P(H_1 \mid F)}{P(H_2 \mid F)}$$

NESTOR could simply output the value of $P^*(H_1 \mid F)$, but this would give the user little understanding of how each of the findings influences the relative likelihood of the two hypotheses being compared. So instead, it calculates a sequence of values using P^*, one for each finding. The ordered list of findings that was used to display an English translation of the causal model is also used here. Call this list FINDINGS and suppose it contains N findings.[67] The i^{th} finding in FINDINGS is designated as f_i. Also, f_0 indicates the null finding, that is, it is an artificial finding with no information. The following algorithm calculates each value of P^*:

[67] The N findings also include any non-causal findings such as age and sex, which are always placed at the head of the list.

For i from 0 to N do the following:

$$P^*(H_1 \mid f_0 \text{ to } f_i) = \frac{P(f_0 \text{ to } f_i \ \& \ H_1)}{P(f_0 \text{ to } f_i \ \& \ H_1) + P(f_0 \text{ to } f_i \ \& \ H_2)}$$

Each value of P^* represents the addition of another finding to a growing set of findings. Successive P^* values indicate how the addition of f_i changes the relative likelihood of H_1 from what it was when only f_0 to f_{i-1} were considered. Since the FINDINGS list is ordered by the causal precedence of the findings, the P^* values indicate how successive findings in the causal ordering influence the relative likelihood of the two hypotheses.

The P^* values are displayed both graphically and numerically. Since, the hypothesis scores used to calculate them are generally bounded probabilities rather than exact ones, the P^* probabilities also are bounded. The specific calculations for the upper and lower bounds on P^* are shown below:

$$P^*_{min}(H_1 \mid f_0 \text{ to } f_i) = \frac{P_{min}(f_0 \text{ to } f_i \ \& \ H_1)}{P_{min}(f_0 \text{ to } f_i \ \& \ H_1) + P_{max}(f_0 \text{ to } f_i \ \& \ H_2)}$$

$$P^*_{max}(H_1 \mid f_0 \text{ to } f_i) = \frac{P_{max}(f_0 \text{ to } f_i \ \& \ H_1)}{P_{max}(f_0 \text{ to } f_i \ \& \ H_1) + P_{min}(f_0 \text{ to } f_i \ \& \ H_2)}$$

6.2. The CRITIQUE Command

The CRITIQUE command provides the user with an indication of how well a single hypothesis accounts for the current clinical findings. Figure 6-5 shows an example of its use in critiquing one of the hypotheses from Figure 6-2.[68] The CRITIQUE command is very similar to the COMPARE command. The main difference is that the critiqued hypothesis is being compared against *all* other hypotheses rather than just compared against one other hypothesis. In terms of implementation, the key difference this makes is in the calculation of

[68]This example is taken from Chapter 2 where the averages of the bounds of probabilities were used for illustrative purposes.

```
* CRITIQUE USER1

THE FOLLOWING CAUSAL MODEL EXPLAINS THE MECHANISMS WHEREBY THE
USER-HYP1 HYPOTHESIS CAUSES THE FINDINGS:

 PRIMARY HYPERPARATHYROIDISM IS CHARACTERIZED BY THE PRESENCE OF INCREASED
 PTH. INCREASED PTH CAN LEAD TO AN INCREASED TOTAL SERUM CALCIUM. THIS
 IN TURN IS CAPABLE OF PRODUCING A DECREASED LEVEL OF CONSCIOUSNESS.
```

FINDING	P(USER1 | F1 TO Fn) NUMERIC FORM (PERCENT)	GRAPHIC FORM (PERCENT) 0 100
F1. NO FINDINGS AVAILABLE	.075	*
F2. SEX IS FEMALE	+0 TO 1	*
F3. TOTAL SERUM CALCIUM IS 13.5	6 TO 16	***
F4. LEVEL OF CONSCIOUSNESS IS COMA	0	*

Figure 6-5: An Example of Critiquing an Hypothesis

the denominator of the equation in Figure 6-4. Instead of being just the sum of the score of two hypotheses, it must be the sum over *all* hypotheses as shown in Figure 6-6.

$$P(H_i \mid F) = \frac{P(F \,\&\, H_i)}{\sum_{j=1}^{N} P(F \,\&\, H_j)}$$

Figure 6-6: A formula for Calculating the Posterior Probability of an Hypothesis

The denominator of this equation is summed over all possible hypotheses and is equal to P(F). The chief problem in implementing the critiquing command is calculating P(F). One of the original reasons for using P(F & H) as a scoring metric was that hypotheses could be

ordered according to their likelihood without the computationally expensive calculation of P(F). The reason that the exact calculation of P(F) is so difficult is that it involves summing over *every* possible hypothesis. For 100 diseases, when all disease combinations are considered, this leads to over 10^{30} possible hypotheses. Clearly, this is not a computationally tractable approach. NESTOR avoids this problem by abandoning the goal of a precise calculation of P(F) and instead calculates its upper and lower bounds.

6.2.1. Calculating the Lower Bound of P(F)

The lower bound of P(F) is found by summing the scores of a subset of the universal set of all possible hypotheses.[69] This is shown in the following equation, where S is the set of all possible hypotheses, and T is a subset of S:

$$P(F) = \sum_{j \epsilon S} P(F \,\&\, H_j) \geq \sum_{j \epsilon T} P_{LB}(F \,\&\, H_j)$$

The subset T is constructed using a modification of the algorithm used for hypothesis formation (see Section 5.4). The major modifications to this algorithm are that no hypotheses are pruned, and each hypothesis is scored as it is generated. Thus, the idea is to use a best-first search strategy that incrementally sums the scores of hypotheses added to T. The better the best-first search strategy, the faster the rate at which the lower bound of P(F) increases with the addition of each new hypothesis to T. Currently, NESTOR adds hypotheses to T according to their prior probabilities, that is, the most *a priori* probable hypotheses are added first. The criteria for halting the construction of T will be discussed shortly (see Section 6.2.3).

6.2.2. Calculating the Upper Bound of P(F)

There are two methods which can be used for calculating the upper bound, depending on circumstances which are discussed below.

---

[69] The hypotheses in the universal set must be mutually exclusive, as they are in NESTOR.

6.2.2.1. Method 1

The upper bound of P(F) is expressed in the following equation:

$$P(F) \leq \sum_{j \epsilon T} P_{UB}(F \,\&\, H_j) + UB\left[\sum_{j \epsilon S-T} P(F \mid H_j) \times P(H_j)\right]$$

The first sum is taken over the upper bounds of the scores of hypotheses that have already been enumerated; Section 6.2.1 discussed how these hypotheses can be generated and their scores calculated. The second sum is taken over the scores of all hypotheses that have not yet been enumerated or scored. Its value is not directly available, but there is a way to determine its upper bound. Assigning the trivial upper bound of one to $P(F \mid H_j)$ results in the following equation:

$$P(F) \leq \sum_{j \epsilon T} P_{UB}(F \,\&\, H_j) + UB\left[\sum_{j \epsilon S-T} P(H_j)\right]$$

The second term is now the upper bound on the prior probability of all hypotheses other than the ones in set T that have already been scored. The second sum can be converted to the difference of two sums and the following inequality results:

$$P(F) \leq \sum_{j \epsilon T} P_{UB}(F \,\&\, H_j) + UB\left[\sum_{j \epsilon S} P(H_j) - \sum_{j \epsilon T} P(H_j)\right]$$

The upper bound of the difference of the two sums can be calculated by taking the upper bound of the first sum and subtracting the lower bound of the second sum, resulting in the following inequality:

$$P(F) \leq \sum_{j \epsilon T} P_{UB}(F \,\&\, H_j) + UB\left[\sum_{j \epsilon S} P(H_j)\right] - LB\left[\sum_{j \epsilon T} P(H_j)\right]$$

Taking 1.0 as the trivial upper bound on the sum involving the universal set S results in the following:

$$P(F) \leq \sum_{j \epsilon T} P_{UB}(F \,\&\, H_j) + 1 - \sum_{j \epsilon T} P_{LB}(H_j)$$

If the upper bound on P(F) is greater than 1.0, then it is set equal to 1.0. Notice that

the upper bound on P(F) does not assume the exhaustiveness of the hypotheses in the set T. Also, because the term

$$UB[\sum_{j \in S} P(H_j)]$$

was set to the maximal value of 1.0, it does not even assume the exhaustiveness of the set of all the hypotheses that can possibly be derived from combinations of diseases *known* to NESTOR (call this the set S'). This also holds true for the lower bound calculated in Section 6.2.1. Therefore, using these bounds on P(F) results in bounds on the posterior probability of the hypothesis in Figure 6-6 that do not assume the completeness of the currently known disease set, that is, S' need not equal S.

6.2.2.2. Method 2

The previous method has the advantage that no assumptions are made about the exhaustiveness of NESTOR's hypothesis space. It has the disadvantage that the upper bound may be close to one if the lower bounds of the prior probabilities of the hypotheses in set T are very small. Also, it is sometimes desirable to know the posterior probability of an hypothesis only with respect to a *particular* set of diseases. In this case tighter bounds than the one given by the previous method are desirable. This section discusses a second method that can sometimes achieve tighter upper bounds.

The second method requires a vector to be called UB.PRIORS. The i^{th} element of UB.PRIORS contains the upper bound on the prior probability of any hypothesis of size N.[70] UB.PRIORS must be precomputed by some method that restricts the hypothesis set to those diseases that the user currently wants included.[71] If there are a total of D diseases, then the largest hypothesis is size D. Now, suppose that set T from Section 6.2.1 contains all

---

[70] The size of an hypothesis equals the number of diseases in that hypothesis.

[71] For example, if the diseases are conditionally independent, they may be ordered by the upper bound of their prior probabilities, from greatest to least; call this list of sorted upper bounds SORTED.DX.PRIORS. Thus, the upper bound of the prior probability of an hypothesis of size N is equal to the product of the first N numbers on SORTED.DX.PRIORS. If some of the diseases are not conditionally independent, then some other method must be used (see Section 4.3.3.2).

hypotheses from size 1 to M, and perhaps some of size M + 1.[72] Then, if COMB(A, B) is the number of combinations of A taken B at a time, the following equation holds:

$$P(F) \leq \sum_{j \epsilon T} P_{UB}(F \& H_j) + \prod_{j=M+1}^{D} UB.PRIOR(j) \times COMB(D, j)$$

The first sum is the same as in Method 1. The second sum calculates the upper bound on the prior probability of every hypothesis not in T. It does this by assigning the upper bound of any hypothesis H of size j to be the maximum upper bound for any hypothesis of size j. The success of this method depends on M being sufficiently large so that the prior probability of any hypotheses greater than size M is very small. Since the prior probabilities of hypotheses generally decrease exponentially as a function of size, this may often be true.

6.2.3. Goal-Oriented Tightening of Bounds on P(H | F)

The previous section discussed two methods for bounding P(F). As more of the hypotheses in the universal hypothesis set S are included in the subset T, the bounds on P(F) become tighter. This leads to tighter bounds on the posterior probability, P(H | F), which is used by the critiquing task. The user is given the option of specifying the maximum acceptable bounds on P(H | F). NESTOR will then continue to add hypotheses to T until either the bounds on P(H | F) are tight enough to meet the user's specifications, or a preset, user-specified time limit is reached.

With slight modification this same method is used to bound the posterior probability of the most probable hypothesis found during hypothesis generation (see Section 5.5.7). In searching for the most probable hypothesis the sum of hypothesis scores in set T is maintained just as above. But, unlike the above methods, during hypothesis search hypotheses are continually being pruned. When there are no hypotheses remaining to be expanded, NESTOR knows that it has discovered the provably most probable hypothesis. However, the most probable hypothesis may have bounds on its posterior probability that are too wide to meet the user's specifications. Since there are no remaining hypotheses to

---

[72] This can be achieved by having the hypotheses enumerated by a best-first strategy that is based on the number of elements in an hypothesis.

expand, there is no way to add hypotheses to set T. The solution to this problem is to maintain a list of the pruned hypotheses. Once the most probable hypothesis has been proven, its posterior probability can be tightened (if necessary) by placing the hypotheses on the pruned hypothesis list (and their expansions) into set T using the previous techniques in this chapter.

6.2.4. The Source of Imprecision in P(H | F)

There are three sources that influence the imprecision (i.e., bounds) of a posterior probability that is calculated by NESTOR.

1. The calculation of the score of hypotheses involves using ***probabilistic knowledge that is bounded*** (see Chapter 3). The source of these bounds is often the experts who construct the knowledge-base. It is therefore difficult to remove this source of imprecision. However, the user is allowed to experiment with its removal by instructing NESTOR to use the averages of the bounds. The user is also allowed to narrow or widen any particular conditional or a prior probability in NESTOR's knowledge-base.

2. The calculation of an hypothesis score also involves using knowledge-based techniques which depend on ***bounding joint conditional probabilities*** (see Chapter 4). This allows NESTOR, when necessary, to maintain the validity of the score by decreasing its precision. This source of imprecision can be lessened by having NESTOR adopt a more radical set of assumptions for aggregating probabilities. However, the resulting increase in precision may be at the expense of the validity of the posterior probability.

3. This chapter has shown that in general NESTOR can only ***bound P(F)***. Since P(F) is integral to NESTOR's calculation of a posterior probability, it is another source of its imprecision. The imprecision due to P(F) can be decreased by having NESTOR consider additional hypotheses in the calculation of P(F). Thus, precision can be increased at the expense of additional computation time.

6.3. Related Work

Section 1.4.1 gives an overview of the explanation capabilities of some medical diagnostic programs from a MDSS viewpoint. This section concentrates on previous work relating specifically to causal or quantitative explanations. The coverage here will not be exhaustive, but will discuss research that is exemplary of the current state of research.

NESTOR is designed to explain the likelihood that a *given* hypothesis causally accounts for a *given* set of findings. Currently, NESTOR does not suggest *additional* findings to seek. Thus, it is not yet programmed to explain the reasons *why* it would be useful to determine the value of a particular finding. Instead, NESTOR focuses on explaining *how* an hypothesis can account for a given set of findings; the remainder of this section will discuss some previous research that has addressed this problem.

In critiquing an hypothesis, NESTOR uses several methods to bound P(F). Similar techniques were independently developed by Horning in the area of grammatical inference [Horning 69]. Section 5.6.2 gives a general orientation to this research and how it relates to NESTOR.

Patil's ABEL program uses a multilevel causal model in the diagnosis of acid-base disorders [Patil 81]. It constructs a case-specific causal model and uses this for scoring an hypothesis as well as explaining the hypothesis to the user. The causal model is translated into English using a method primarily developed by Swartout [Swartout 81]; NESTOR uses a similar translation method. ABEL contains five levels of causal detail in its case-specific models, in contrast to NESTOR's two levels. This allows ABEL to provide any one of five different levels of causal detail in its explanations, in contrast to Nestor's two levels; other than this, the *qualitative* explanation produced by NESTOR is very similar to that of ABEL.

The primary difference in the explanation produced by NESTOR and ABEL is that NESTOR presents a *quantitative assessment* of the likelihood of a given hypothesis with respect to a given set of findings. ABEL contains no representation of uncertainty in its causal links, and therefore can present only a qualitative discussion of the causal relationships that exist among the findings. In contrast, NESTOR contains knowledge of

the probabilities associated with the causal links and it uses these to produce a graph that illustrates how each finding contributes to the score of an hypothesis.

Wallis and Shortliffe have developed a program which is like ABEL in the sense that the causal explanation it produces is purely qualitative [Wallis 82]. They have refined the process of explaining a causal model by explicitly representing the notion of the complexity of a causal node or link. This allows their program to tailor an explanation on the basis of the user's expertise and desire for details of the causal interactions. Currently, NESTOR does not contain a model of the user. Section 6.4.1 discusses the advantages of such a model.

Spiegelhalter and Knill-Jones have developed a medical diagnostic program which is based on formal probability theory [Spiegelhalter 84]. These researchers argue that a formal probabilistic system is as capable as a nonformal one, such as those commonly found in AI medical diagnostic programs, in explaining the reasoning behind its diagnosis. The program they have developed is able to quantitatively indicate how each finding in a given case contributes to the total probability of an hypothesis. The method they use for quantitative explanation is similar to NESTOR's probability graph. Both programs communicate the impact of each finding on the probability of a given hypothesis. However, their program does not represent the causal relationships between findings. Thus, the pathophysiological mechanisms that link the findings can not be included in an explanation. These mechanisms, when they are known, constitute an important part of clinical reasoning, and their presence in an explanation is of prime importance. This is the major reason for NESTOR's incorporating causal knowledge into its explanation.

NESTOR's explanation method is similar to a combination of Patil's method and the method of Spiegelhalter and Knill-Jones. It produces a qualitative causal explanation of the interrelationships of findings like Patil's program, and is capable of doing this at more than one level of detail. Like the program of Spiegelhalter and Knill-Jones, NESTOR produces a quantitative assessment of how each finding influences the probability of an hypothesis. The qualitative causal explanation serves as a general context within which NESTOR's quantitative assessment can be interpreted. In terms of the individual components of its explanation, NESTOR does not currently offer anything radically new. The primary contribution of its approach is the emphasis on the importance of providing both the qualitative and the quantitative facets of causal explanation.

NESTOR's current explanation capability is only a start. The majority of the research so far has been in developing a formal foundation for future research in explanation. There is still a significant amount of research that is needed to discover how to synthesize qualitative and quantitative causal explanation. The next section proposes several ways of doing this.

6.4. Extensions

6.4.1. Tailored explanations

NESTOR could provide better explanations if it knew more about the complexity of its knowledge and the expertise and needs of its users. Some research into these areas is already in progress. The previous section discussed work by Wallis and Shortliffe in which causal knowledge is labeled according to its complexity [Wallis 82]. The users explicitly inform the program about their level of expertise and it then generates an explanation that is appropriate for that level. Clancey and later London incorporated a user model into the explanation component of a medical tutorial system called GUIDON [London 82, Clancey 79]. The user model is updated as more information about the user is discovered in the course of an interaction. The advantage of such a model is that explanations can be *automatically* tailored to the user's level of expertise. Once the medical knowledge is categorized (e.g., complexity labeling in the program of Wallis and Shortliffe) and the user's level of expertise is known (e.g., using a user model as in GUIDON), then the information to include in the explanation can be determined.

The above areas of research are important to good explanation and NESTOR could certainly improve its explanation facility by incorporating and extending them. Since NESTOR represents causal relationships between findings probabilistically, it is ideally suited for tailoring explanations on the basis of the probabilistic *significance* of individual findings. For example, when CRITIQUING an hypothesis NESTOR might only discuss those findings which *significantly* increase or decrease the probability of the hypothesis (with respect to all other hypotheses).

Finally, more psychological research is needed to understand the type of explanations

that physicians desire. Weiner has investigated aspects of this issue in developing a program called BLAH [Weiner 80]. However, there is still much that remains unknown. For example, in the case of NESTOR, will its use of a numerical probability graph be acceptable, or will most physicians prefer to see this information expressed using qualitative prose? It is certainly possible that different physicians will desire different formats of presentation of the same information. The investigation of such psychological issues will significantly aid the future development of explanation research in computer-aided medical decision-making.

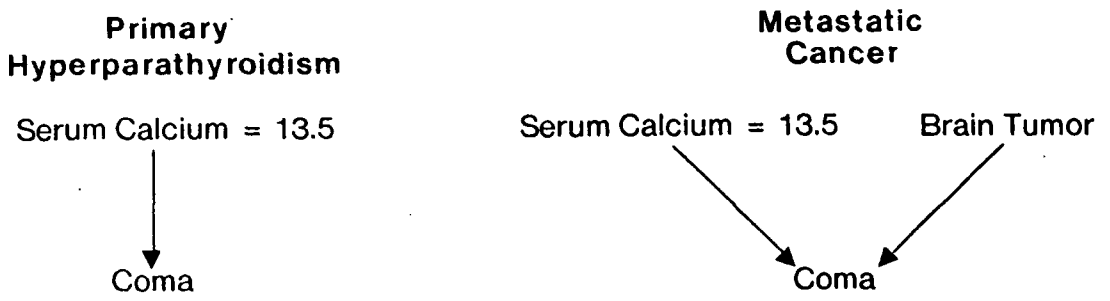
6.4.2. An Analysis of the Local Causal Factors Affecting the Relative Likelihood of Two Hypotheses

It would be helpful to explain *why* the relative probabilities of the hypotheses change as each finding is introduced. For example, at the bottom of Figure 6-2 is a probability graph showing how the likelihood of primary hyperparathyroidism changes relative to metastatic cancer as each finding is considered. After coma is added to the list of findings, the probability of primary hyperparathyroidism drops to zero relative to metastatic cancer. In order to understand why this occurred, the user must refer to the qualitative causal graphs of the two hypotheses and combine this with knowledge of the quantitative significance of each causal link. An improvement in the program would be for NESTOR to analyze the causal links that have a quantitatively significant influence on the relative probabilities of the two hypotheses and to explicitly report these to the user as an adjunct to the probability graph. For the example in Figure 6-2 additional explanation might appear as follows:

```
F1. The prior probability of metastatic cancer is greater
    than that of primary hyperparathyroidism.

F2. The sex of the patient being female slightly favors
    the diagnosis being primary hyperparathyroidism as
    opposed to metastatic cancer.

F3. A total serum calcium of 13.5 mg/100ml does not
    significantly favor either of the two hypotheses.

F4. The fact that the patient is in a coma greatly favors
    metastatic cancer over primary hyperparathyroidism.
    This is because primary hyperparathyroidism can only
```

```
cause coma by way of an increased serum calcium, yet the
current serum calcium of 13.5 mg/100ml is insufficient
to cause coma. However, metastatic cancer is also able
to cause coma by way of a metastatic brain tumor. Thus,
metastatic cancer can account for all the findings,
whereas primary hyperparathyroidism can not.
```

The key to these type explanations lies in analyzing the *local* causal links that are responsible for the scores of the two hypotheses being compared. In the example, the F4 explanation was derived from the following local causal links:



The F4 explanation is derived by first structurally comparing the local causal factors that differentiate the two hypotheses once the finding of coma is introduced. In this case, the structural difference is due to metastatic cancer having an additional link to coma via brain tumor. Next, the probabilities of the links are assessed to determine which of the local structural differences significantly contributes to the relative change in the score of the two hypotheses. In this case, a serum calcium of 13.5 causes coma with a probability of zero, whereas a brain tumor causes it with a probability between 0.1 and 10 percent. Thus, this accounts for the disparity in the scores of the two hypotheses.

6.4.3. Extensions to the CRITIQUE Command

Section 6.4.2 has outlined one possible extension to the COMPARE command. The CRITIQUE command can be extended in a manner that is similar but more complex than that of COMPARE command. The COMPARE command only compares the local causal interactions of two hypotheses, whereas the CRITIQUE command must compare the causal interactions of one hypotheses with respect to *all* other hypotheses. The difficulty arises in

summarizing all the possible differences. There are several factors and techniques that could help in this regard. First, the hypothesis being critiqued could be compared only to the most probable alternative hypotheses. Second, it is possible that many of the alternative hypotheses will share the same local differences in causal structure with respect to the hypothesis being critiqued. Third, the differences, if widely different, could be abstracted to some level at which they are similar. There are no doubt other methods for extending the CRITIQUE command, and this seems to be a promising area for future explanation research.

Chapter 7

Summary and Conclusions

NESTOR is composed of three program modules which address several existing problems in computer-aided medical decision making involving hypothesis searching, scoring, and explanation. Figure 7-1 shows how these tasks relate to the larger task of medical decision-making. This chapter summarizes for each of the tasks the technical problems addressed by NESTOR, the methods used to address these problems, the success of the methods in solving the problems, and the limitations of the work which help to define directions for future research. Finally, the strengths and limitations of NESTOR as a unified medical decision support system are discussed.

7.1. Scoring a Diagnostic Hypothesis

7.1.1. Problems, Methods, and Results

The Scoring Metric

Every diagnostic program must have some means of determining the relative likelihood of hypotheses. Typically, a score is assigned to each hypothesis and the one with the highest score is considered the most likely.[73] In this regard, many previous programs have used an ad hoc scoring scheme that has no straightforward formal interpretation. On the other hand, implementations of programs that are based on formal probability theory commonly introduce assumptions that may be invalid.

---

[73]Cohen has recently experimented with categorical methods for rank ordering hypotheses, but the generality of this approach is not known [Cohen 83]. Given the highly probabilistic nature of knowledge in medicine it seems unlikely that purely categorical methods will be adequate for general medical diagnosis.

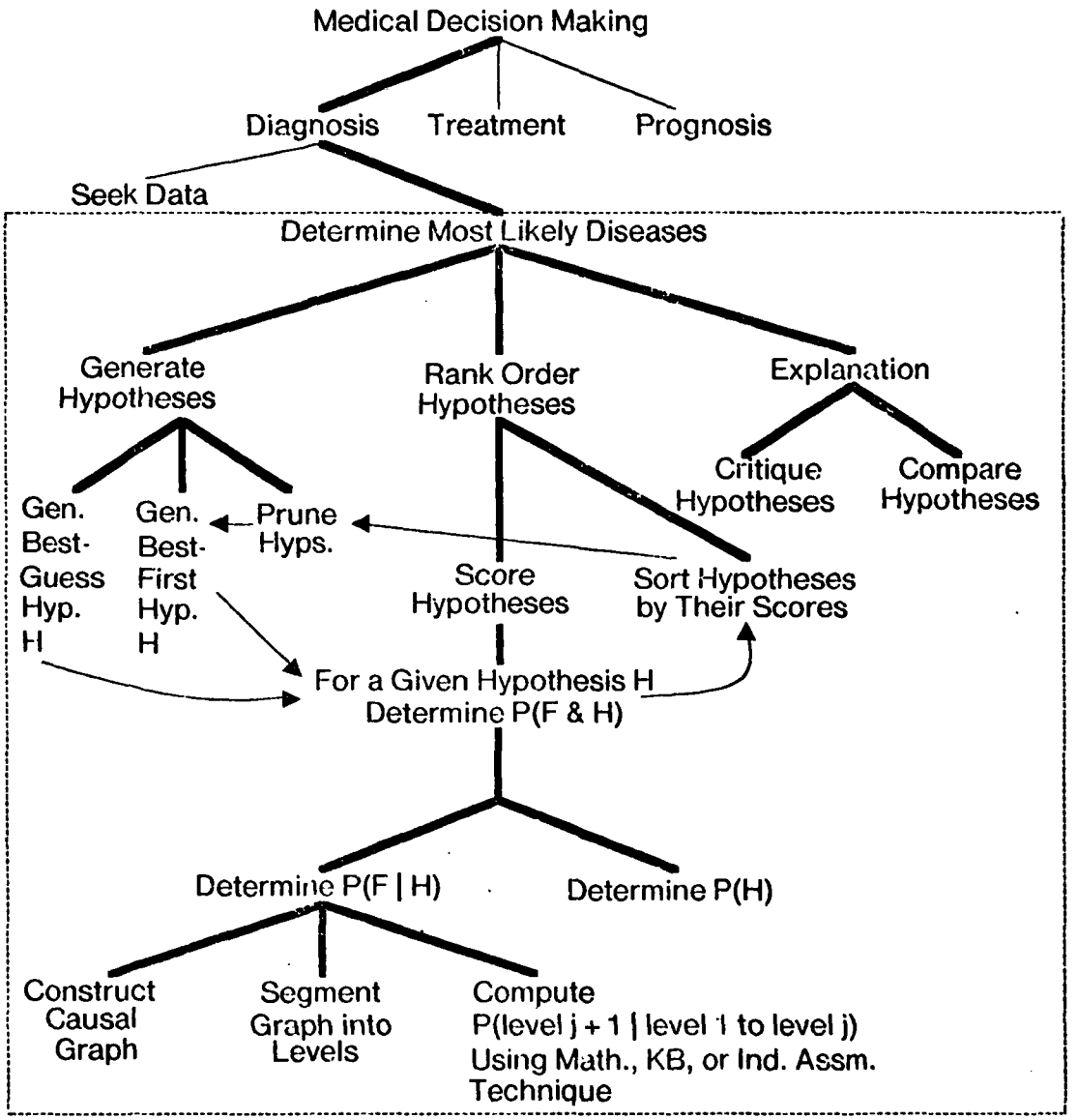


Figure 7-1: A Summary of the Tasks Performed by NESTOR

NESTOR uses P(F & H) as the score of an hypothesis H given a set of findings F. [74] P(F & H) has the advantage of being a formal probability with a clear interpretation. This makes it relatively easy to communicate the goal of the scoring process to other researchers, and to interface this scoring module to other independently developed decision making modules. I also found that having a precisely defined probability as a scoring metric was a significant aid in designing the details of the scoring algorithm. This was largely due to probability theory being a well developed field of mathematics upon which I could use previously developed definitions, relations, techniques, and theorems. An ad hoc scoring scheme would have provided none of this conceptual support. The efficiency and insight gained from working with a well developed theory should not be underestimated.

In addition to being probabilistically formal, the scoring metric also allows the most likely hypothesis to be located without incurring the assumptions of hypothesis mutual exclusivity and exhaustiveness that are often made in implementing formal probabilistic systems based on Bayes' formula. However, the disadvantage of P(F & H) as a scoring metric is that even though the most likely hypothesis can be determined, its posterior probability remains unknown. In cases where the most probable hypothesis H is still very improbable, knowing H, but not its probability given the findings, can be misleading. Section 7.3.1 discusses some methods that were developed to address this problem.

Bounding Probabilities

One of the problems with most diagnostic systems is that the uncertainty of probabilistic statements is not represented. Often the precise probability of a finding given a disease is not known, but such systems require that the probability be entered into the knowledge base as a single, precise number. Thus, the original uncertainty (imprecision) in the probability is artificially lost.

NESTOR's approach to this problem is to bound all probabilities. The expert who helped me develop NESTOR's knowledge-base reported that the ability to bound

---

[74] Section 5.3 proves that this scoring metric is adequate to partially rank order hypotheses according to their likelihood given the findings.

probabilities made the task of knowledge acquisition significantly easier and more natural than if a single probability were required.

The use of bounds has also significantly increased the flexibility in designing the hypothesis scoring techniques. It is no longer necessary either to know all the relevant probabilities or to make strong assumptions, as is commonly done in implementing Bayes' formula. Instead, the lack of complete probabilistic knowledge is reflected in the imprecision of the resulting hypothesis score without sacrificing the validity of the score. Furthermore, other knowledge sources, such as causal knowledge, can be used to augment sparse probabilistic knowledge so that the tightness of the bounds can be increased without jeopardizing the validity of the score; this is discussed next.

Integrating Causal and Probabilistic Knowledge

One of the problems in taking a strictly probabilistic approach to decision making is the lack of a sufficient corpus of probabilities. Typically medical diagnostic systems address this problem by using strong assumptions in their scoring procedure. Such assumptions allow a small set of probabilities to be applied in a large number of cases. An example of this is the application of Bayes' formula which assumes conditional independence of the findings. The problem with such approaches is that their assumptions may be invalid and lead to diagnostic errors [Norusis 75] (see Section 1.5.1.1). The previous section discussed how NESTOR uses bounds on probabilities to maintain the validity of a score (while possibly sacrificing precision). The calculation of the bounds is always relative to some set of assumptions. Without additional knowledge, a moderately conservative set of assumptions applied to a sparse set of probabilities will often yield wide bounds on the scores of hypotheses. NESTOR has been used to investigate methods of narrowing these bounds by using causal knowledge to guide the combination of probabilities. The chief assumptions introduced are that the causal knowledge is valid and that all important causal interactions[75] between findings are represented in the knowledge-base.

---

[75] An *important* causal interaction is defined as one which if not represented would lead to invalid scoring of some hypothesis given some set of findings. Note that NESTOR does *not* require that the detailed mechanisms *between* all findings be known.

NESTOR uses causal knowledge to structure the dependency of findings. This allows it to correctly combine the conditional probabilities that link findings into the joint conditional probability of the findings which is needed for scoring an hypothesis. The causal structure is an acyclic directed graph in which each node is a multivalued finding or intermediate causal node.[76] A causal simulation algorithm has been developed which provides a general means of properly using this causal structure to score an hypothesis. An evaluation showed that the causal structuring of knowledge in NESTOR was critical in permitting it to correctly diagnose a number of cases (see Section 4.5). Although NESTOR is designed to use causal knowledge in scoring diagnostic hypotheses, the absence of such knowledge will lead it to perform no worse than a Bayesian program that assumes conditional independence of the findings. Thus, valid causal knowledge only serves to improve the accuracy of diagnosis in NESTOR.

Within a particular causal graph it is necessary to determine the probability of a set of local effects given their immediate causes. We have termed this the calculation of local joint conditional probabilities (local JCP's). NESTOR is able to use a number of methods in calculating local JCP's. These methods are based on different explicit sets of assumptions, which the user can optionally select. Like most AI programs, NESTOR separates the domain knowledge from the inference procedure. However, it also goes one step further in separating the inference assumptions from the primary inference procedure. This allows inference assumptions to be dynamically selected based on current goals and domain knowledge.

The most intricate method for calculating local JCP's involves using categorical causal knowledge and default reasoning in calculating the bounds of local JCP's. An evaluation indicated that this relatively complex method made no difference in the quality of the resulting diagnosis relative to a simple method that assumed that effects are conditionally independent given their immediate causes. This suggests that Bayes' formula with the assumption of independence can be validly applied at the local level (given the causal structuring of the findings) although not at the global level (that is, in the absence of a

---

[76] The ability to represent multivalued variables allows varying degrees of finding severity to be represented in a natural way.

causal structuring of the findings). Put another way, the experiments in this dissertation suggest that the scoring power of causal knowledge lies in its globally structuring the dependency of findings and not in locally directing their combination (see Section 4.5.2.2.2). The generality of this result will require further investigation.

The evaluation also suggested that rank ordering hypotheses by their upper bounds was sufficient in most cases for diagnosis (see Section 4.5.3). The use of upper bound scores is compatible with a strategy in which an hypothesis is not ruled-out until it is provably poor (i.e., its upper bound score is low or zero). Thus, this provides a conservative diagnostic strategy, which is capable of using known probabilities and categorical causal knowledge to affect the ranked order of an hypothesis in the differential diagnosis list.

Multiple Disease Hypotheses

In complex medical cases it is important to be able to consider multiple disease hypotheses. One problem this creates involves integrating the effect of multiple etiologies on the probability of occurrence of a set of findings. NESTOR is able to use its causal knowledge to integrate multiple diseases into a coherent causal graph which explicitly represents the interactions of multiple etiologies. An evaluation showed that this causal representation of multiple disease interaction can significantly improve the quality of the scores of multiple disease hypotheses (see Section 4.5.2).

7.1.2. Limitations and Future Research

Although the use of causal knowledge aided in tightening the bounds of local JCPs, the scores of many hypotheses were often overlapping. This was typically due to hypotheses having a lower bound score of zero. Thus, in the evaluation of NESTOR's scoring method it was decided to use the upper bound as the scoring metric. This method worked well in all but one case (see Section 4.5.2.2.1). The initial success of this method suggests that other methods of using knowledge to minimize upper bounds on scores should be investigated with the thought of using the upper bound score as a scoring metric.

Currently, NESTOR can only represent probabilities between a cause and an effect. A more general representation would allow the expression of conditional probabilities between any nodes regardless of their causal relationship. For example, this would allow the correlation between two findings to be expressed, even if one did not cause the other.

Time is not explicitly represented in NESTOR presently, nor is the closely related phenomenon of causal feedback. The principle difficulties in using time stem from the large number of possible temporal probabilities, their scarcity in the literature, and the difficulty of estimating them from clinical experience. However, time must eventually be incorporated into NESTOR if general diagnosis and the proper handling of causal feedback are to be accomplished. Section 4.6.6 suggests several ideas for addressing this problem. Other suggestions for future research are discussed in Section 4.6.

7.2. Searching for the Most Probable Hypothesis

7.2.1. Problems, Methods, and Results

NESTOR's ability to consider multiple disease hypotheses when searching for the most likely hypothesis creates a very large hypothesis search space; the number of possible hypotheses is an exponential function of the number of diseases in the knowledge-base. A branch and bound search technique based on the prior probability of hypotheses was developed in order to decrease the size of this space. An evaluation of 50 test cases showed that the technique could on average reduce the search time required to locate the most probable hypothesis from 200 seconds[77] to less than 20 seconds.

As with the design of NESTOR's scoring method, the design of its search method was greatly aided by having a formal probability as a scoring metric. This allowed well known properties of probability theory to be applied in designing the search algorithm and in proving that it is admissible. It also facilitates the communication of this method to other researchers.

---

[77] This is the average amount of time required to locate the most probable hypothesis by exhaustively searching an hypothesis space that consists of all combinations of the 7 diseases known to NESTOR.

With regard to flexibility, NESTOR gives the user significant power to tailor its search for the most probable diagnostic hypothesis. As examples, the disease set can be restricted, as can the search time, and the maximum number of diseases per hypothesis can be specified, as can number of top hypotheses to be located. Although no formal evaluation was made, the examples in Chapter 2 suggest that these features do enhance the usefulness and acceptability of NESTOR as a decision making aid.

7.2.2. Limitations and Future Research

NESTOR's search algorithm proved practical in terms of its search time when applied to 50 test cases[78] However, a general analysis indicated that even with this technique the search time will increase exponentially as a function of the number of findings and the number of diseases in the best diagnosis. Section 5.8 suggests ways in which the rate of exponential growth can be curbed. One method involves using causal knowledge to prune the search, while a second method involves searching a smaller, more abstract hypothesis space.

7.3. Explanation

7.3.1. Problems, Methods, and Results

NESTOR provides two explanation commands: the COMPARE command and the CRITIQUE command. The COMPARE command uses both qualitative causal knowledge and quantitative probability knowledge in comparing two hypotheses. The CRITIQUE command is similar in that it abstractly compares a single hypothesis to all other hypotheses. Chapter 2 gives several examples of the applications of the two commands in which they appear to produce useful explanations.

The implementation of the CRITIQUE command led to the development of a method for validly bounding the posterior probability of an hypothesis without exhaustively enumerating all hypotheses, which is often assumed to be necessary. The key to the success

---

[78] These cases contained 12 or fewer findings with a knowledge-base of 7 diseases.

of this approach is that its goal is merely to *bound* the posterior probability rather than attempt to calculate it exactly. The technique has the desirable property that the bounds become progressively tighter with the increased expenditure of calculation time.

The COMPARE and CRITIQUE commands exemplify the essence of the decision support system concept in which the user is given significant control of the interaction with the computer. With these commands the user is free to choose which hypotheses are to be compared or critiqued and at which time. These two commands, in conjunction with the user's freedom to enter particular hypotheses and NESTOR's hypothesis generation capability, provide the user with a great deal of flexibility and power in exploring the most likely causes of a patient's illness.

7.3.2. Limitations and Future Research

Additional research is needed in developing the ability to provide a more integrated discussion of the causal factors that significantly affect the score of an hypothesis. Currently, a qualitative causal overview is given followed by a quantitative assessment of how each finding affects the overall score of an hypothesis. However, the burden is on the user to integrate these two stages of explanation. A better explanation would result if NESTOR explicitly indicated to the user the local causal conditions that account for the impact of a given finding on the score of an hypothesis (see Section 6.4.3).

7.4. NESTOR as a Medical Decision Support System

The three previous sections of this chapter discussed the technical methods used in NESTOR to create a medical decision support system (MDSS). The remainder of this chapter discusses the current status and future development of NESTOR as a unified MDSS tool.

A user's interaction with NESTOR is coordinated by an interface that provides a decision support system environment in which the user is given significant control of the problem solving process. This environment is more of a tool for interactively exploring diagnostic possibilities than a black box which receives patient data and outputs diagnostic

labels. In particular, NESTOR allows users to explore interactively their own diagnostic hypotheses. In addition, users can seek NESTOR's diagnostic opinion and interactively examine how it compares to their own. The nature of this interaction is akin to the type that occurs between an expert physician consultant and a physician seeking advice. An important characteristic of this type of exchange is that it creates a dialog in which dogmatic advice is avoided. This is significant because dogmatic advice is unlikely to be tolerated by physician users.

Currently, NESTOR is able to generate and evaluate hypotheses given a set of findings. This is one important component of diagnosis. NESTOR does not currently consider the other major component -- the ability to determine which findings to seek next, as shown in Figure 7-1.[79] This task requires the formulation of a diagnostic strategy, which opens up an entire new area of decision support functions, e.g. the critiquing of a user's diagnostic strategy. Formulating a diagnostic strategy for the acquisition of findings requires that consideration be given to the costs and benefits of proposed tests. Thus, this extension will require that NESTOR consider utilities in addition to the probabilities with which it currently deals.

Figure 7-1 shows that therapy and prognosis are also currently outside the realm of decision support tasks performed by NESTOR. However, there is no reason why an MDSS approach can not be applied to these tasks too. In fact, therapeutic medical decision support is an area of active research, as discussed in Section 1.4.2. The ideal MDSS would of course provide a unified consultation environment encompassing diagnosis, therapy, and prognosis. Although such a system has not yet been achieved, the gradual development and refinement of its components are making this goal ever closer to reality.

NESTOR has been designed primarily as a aid to physicians seeking consultation for difficult cases. However, there is no reason why it could not be used by other healthcare personnel who understand its medical vocabulary. For example, it could be used as an effective teaching aid to medical students. Clinical cases might be provided by the student, prestored in a database, or dynamically generated by NESTOR. Given the findings in the

---

[79] Section 1.2 discusses how these two components are complementary.

case, the student could actively explore diagnostic possibilities by using NESTOR's ability to critique and compare hypotheses. NESTOR's explanations regarding the causal strengths and weaknesses of student hypotheses might be especially instructive.

Currently, NESTOR is designed for use with a standard computer terminal. The patient findings and user hypotheses are entered via a keyboard -- a task that is often very tedious. Facilitating or eliminating keyboard interaction would significantly improve the user interface to NESTOR. Chapter 3 discusses NESTOR's current method for facilitating this process by providing a flexible approach to the entry of findings. However, this still requires that each finding be manually entered. The use of a menu-driven touch screen is an alternative approach that is potentially useful, since it replaces typing with pointing. In the longer term, voice input may also be possible. However, the ideal situation would be to directly interface NESTOR to a clinical database which already contained the findings. Such databases are not yet common, and therefore the practicality of this approach must await more widespread installation of clinical databases.

NESTOR's output uses text and bar graphs. One potentially useful addition would be to graphically display how a diagnostic hypothesis can causally account for the current set of patient findings. If the thickness or color the causal arrows reflected the probability of the effects given their causes, then this would provide a quick visual indication to the user of important areas of the graph on which to focus. The relationships of the causal mechanisms of two hypotheses that are being compared could also be graphically displayed with the use of different colors to indicate the causal links that are shared in contrast to those unshared.

Until now the primary purpose of NESTOR has been to explore a number of research ideas in computer-aided medical diagnosis (as discussed in Sections 7.1, 7.2, and 7.3) with a program that provides a medical decision support environment. In order for NESTOR to move from a research tool to a clinically useful aid, a number of developments will need to take place. Previous sections of this chapter have suggested some possible improvements in the representation and the user-interface. However, probably the most critical area needing development is the knowledge-base. Currently, NESTOR is limited to a small test domain of seven diseases that cause hypercalcemia. This domain has been sufficient as an initial testbed for the ideas being explored. but the demonstration of NESTOR as a clinically useful tool will require a significantly larger knowledge-base.

There are numerous stages in the evaluation of a decision making aid. The following steps of an evaluation have been suggested by Shortliffe and Davis [Shortliffe 75], and a review of them may help place the current status of NESTOR in perspective:

1. *Demonstrate a need for the system.* In order for NESTOR to meet this criteria it will probably be necessary to develop a knowledge-base that is significantly larger than the current one. The addition of a module to strategically seek new findings is probably also necessary for the program to be useful in a real clinical setting where in addition to developing a differential diagnosis the issue is often one of deciding what to do next.

2. *Demonstrate that the system performs at the level of an expert.* The current use of causal knowledge for improving the accuracy of hypothesis scoring and the use search methods that can diagnosis multiple disease hypotheses are expected to contribute toward improving NESTOR's diagnostic accuracy and thus performance expertise. The precise extent of this contribution will obviously depend on the nature of the medical domain chosen. Chapter 4 illustrates several examples in which causal knowledge is critical to a correct diagnosis within the area of hypercalcemic disorders. Obviously, NESTOR's methods are most useful in domains in which causal knowledge is most prevalent. However, recall that when no causal knowledge is available, NESTOR is capable of behaving like a Bayesian program that uniformly assumes conditional independence. Bayesian programs that assume conditional independence have already been shown to behave at the expert level in some domains [deDombal 74], and NESTOR would do likewise. NESTOR's additional use of valid causal knowledge will only serve to improve its diagnostic accuracy within such domains, as well as other domains where causal interactions are more critical.

3. *Demonstrate the system's useability.* This criteria relates largely to the user interface. The current commands in NESTOR are small in number, straightforward, and easily learned.[80] A major problem with the current interface, and that of almost all computer-aided decision making programs today, is the tedious task of entering findings.

4. *Demonstrate acceptance of the system by physicians.* The decision support system approach of NESTOR should significantly aid in improving physician

---

[80]The command set will of course need to expand as the tasks of data gathering, therapy, and prognosis are developed.

acceptance, regardless of the medical domain of application or future technical improvements that are made to NESTOR. The philosophy is primarily one of avoiding dogmatic advice by allowing the user to share in the problem solving process with the program to the extent desired by the user.

5. *Demonstrate an impact on the management of patients.* Physicians will only be willing to accept advice if they can be convinced that it is good advice. For the same reasons discussed in 4 above, the use of a decision support system approach should positively impact the acceptance of NESTOR's advice and its eventual use in the management of patients.

6. *Demonstrate an impact on the well-being of patients.* This point is difficult to formally demonstrate, but it is obviously the primary goal of any decision making aid. In addition, the aid must also be cost-effective if it is to be used in the real world of limited resources. NESTOR is not yet near a stage of development where a test of its clinical impact would be appropriate. However, it is important to remain conscious that the eventual impact of NESTOR on the well-being of the patient must be the ultimate determinant of its utility.

It is apparent that additional research and development are needed before NESTOR becomes fully operational in a clinical setting. However, the basic principles upon which it was designed appear strong. In particular, its use of a decision support system approach and its application of symbolic causal knowledge within a formal probabilistic framework seem to be powerful concepts that can productively guide future research in computer-aided medical decision making.

References

[Allen 81] Allen, J.F.
A general model of action and time.
Technical Report 86, Department of Computer Science, University of Rochester, 1981.

[Barnett 81] Barnett, J.A.
Computational methods for a mathematical theory of evidence.
In *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, pages 868-875. IJCAI, 1981.

[Barr 81] Barr, A., and Feigenbaum, E.A.
The Handbook of Artificial Intelligence, Volume I.
William Kaufmann, Inc., Los Altos, CA, 1981.

[Berwick 81] Berwick, D.M., Fineberg, H.V., and Weinstein, M.C.
When doctors meet numbers.
The American Journal of Medicine 71:991-998, 1981.

[Buchanan 84] Buchanan, B.G., and Shortliffe, E.H.
Rule-Based Expert Systems.
Addison-Wesley, Reading, Massachusetts, 1984.

[Charniak 83] Charniak, E.
The Bayesian basis of common sense medical diagnosis.
In *Proceedings of the Third National Conference on Artificial Intelligence*, pages 70-73. The American Association for Artificial Intelligence, 1983.

[Clancey 78] Clancey, W.J.
An antibiotic therapy selector which provides for explanations.
Technical Report HPP-78-26, Department of Computer Science, Stanford University, December, 1978.

[Clancey 79] Clancey, W.J.
Transfer of Rule-Based Expertise through a Tutorial Dialogue.
PhD thesis, Department of Computer Science, Stanford University, 1979.
Report no. STA-CS-769.

[Cohen 83] Cohen, P.R.
Heuristic Reasoning about Uncertainty: An Artificial Intelligence Approach.
PhD thesis, Stanford University, 1983.

[Davis 76] Davis, R.
Applications of Meta-Level Knowledge to the Construction, Maintenance, and Use of Large Knowledge Bases.
PhD thesis, Department of Computer Science, Stanford University, 1976.

[Davis 83] Davis, R.
Diagnosis via causal reasoning: Paths of interaction and the locality principle.
In *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, pages 88-94. IJCAI, 1983.

[deDombal 74] deDombal, F.T., Leaper, D.J., Horrocks, J.C., Staniland, J.R., and McCain, A.P.
Human and computer-aided diagnosis of abdominal pain: Further report with emphasis on performance.
British Medical Journal 1:376-380, 1974.

[Detmer 78] Detmer, D.E., Fryback, D.G., and Gassner, K.
Heuristics and biases in medical education.
Journal of Medical Education 53:682-683, 1978.

[Duda 76] Duda, R.O., Hart, P.E., and Nilsson, N.J.
Subjective Bayesian methods for rule-based inference systems.
In *Proceedings of the 1976 National Computer Conference*, pages 1075-1082. AFIPS Press, 1976.

[Fagan 80] Fagan, L.
VM: Representing Time-Dependent Relations in a Clinical Setting.
PhD thesis, Department of Computer Science, Stanford University, 1980.

[Garvey 81] Garvey, T.D., Lowrence, J.D., and Fischler, M.A.
An inference technique for integrating knowledge from disparate sources.
In *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, pages 319-325. IJCAI, 1981.

[Harvey 79] Harvey, C.M.
Operations Research: An Introduction to Linear Optimization and Decision Analysis.
Elsevier North Holland, Inc., New York, 1979.

[Hillier 74] Hillier, F.S., and Lieberman, G.J.
Introduction to Operations Research.
Holden-Day, San Francisco, 1974.

[Horning 69] Horning, J.J.
A Study of Grammatical Inference.
PhD thesis, Department of Computer Science, Stanford University, 1969.
Report no. CS-139.

[Johnson 79] Johnson, P.E., Severance, D.G., and Feltovich, P.J.
Design of decision support systems in medicine: Rationale and principles from the analysis of physician expertise.
In Shriver, B., Walker, T., and Sprague, R., Jr. (editors), *Proceedings of the Twelfth Annual International Conference on System Sciences.* Western Periodicals Company, 1979.

[Kahn 77] Kahn, K., and Gorry, G.A.
Mechanizing temporal knowledge.
Artificial Intelligence 9:87-108, 1977.

[Keen 78] Keen, P.G.W., and Morton, M.S.S.
Decision Support Systems.
Addison-Wesley, Reading, Massachusetts, 1978.

[Komaroff 79] Komaroff, A.L.
The variability and inaccuracy of medical data.
Proceedings of the IEEE 67:1196-1207, 1979.

[Konolige 79] Konolige, K.
Bayesian methods for updating probabilities.
Technical Report, Appendix D, Final Report, SRI Project 6415, Stanford Research Institute International, 1979.

[Langlotz 83] Langlotz, C.P., and Shortliffe, E.H.
Adapting a consultation system to critique user plans.
International Journal of Man-Machine Studies 19:479-496, 1983.

[Leaper 72] Leaper, D.J., Horrocks, J.C., Staniland, J.R., and deDombal, F.T.
Computer-assisted diagnosis of abdominal pain using estimates provided by clinicians.
British Medical Journal 4:350-354, 1972.

[Ledley 59] Ledley, R.S., and Lusted, L.B.
Reasoning foundations of medical diagnosis.
Science 130:9-21, 1959.

[Lee 78] Lee, D.B.N., Zawada, E.T., and Kleeman, C.R.
The pathophysiology and clinical aspects of hypercalcemic disorders.
The Western Journal of Medicine 129:278-320, October, 1978.

[Lemmer 76] Lemmer, J.F.
Algorithms for Incompletely Specified Distributions in a Generalized Graph Model for Medical Diagnosis.
PhD thesis, University of Maryland, 1976.

[Lemmer 82] Lemmer, J.F.
Efficient minimum information updating for Bayesian inferencing in expert systems.
In *Proceedings of the Second National Conference on Artificial Intelligence.* The American Association for Artificial Intelligence, 1982.

[London 82] London, B., and Clancey, W.J.
Plan recognition strategies and student modeling: Prediction and description.
In *Proceedings of the Second National Conference on Artificial Intelligence*, pages 335-338. The American Association for Artificial Intelligence, 1982.

[Ludwig 81] Ludwig, D.W.
INFERNET - A computer-based system for modeling medical knowledge and clinical inference.
In *Proceedings of the Fifth Annual Symposium on Computer Applications in Medical Care*, pages 243-249. IEEE, 1981.

[Ludwig 83] Ludwig, D., and Heilbronn, D.
The design and testing of a new approach to computer-aided differential diagnosis.
Methods of Information in Medicine 22:156-166, 1983.

[McDermott 81] McDermott, D.
A temporal logic for reasoning about processes and plans.
Technical Report 196, Department of Computer Science, Yale University, 1981.

[Miller 77] Miller, M.C., III, Westphal, M.C., Reigart, J.R., and Barner, C.
Medical Diagnostic Models: A Bibliography.
University Microfilms International, Ann Arbor, Michigan, 1977.

[Miller 82] Miller, R.A., Pople, H.E., Jr., and Myers, J.D.
INTERNIST-I, an experimental computer-based diagnostic consultant for general internal medicine.
The New England Journal of Medicine 307:468-476, 1982.

[Miller 83] Miller, P.L.
Critiquing anesthetic management: The ATTENDING computer system.
Anesthesiology 53:362-369, 1983.

[Mittal 82] Mittal, S.
Event-based organization of temporal databases.
In *Proceedings of the Fourth National Conference of the Canadian Society for Computational Studies of Intelligence.* Canadian Information Processing Society, May, 1982.

[Myers 77] Myers, W.P.L.
Differential diagnosis of hypercalcemia and cancer.
CA - A Cancer Journal for Clinicians 27:259-272, 1977.

[Norusis 75] Norusis, M.J., and Jacquez, J.A.
Diagnosis I. Symptom nonindependence in mathematical models for diagnosis.
Computers and Biomedical Research 8:156-172, 1975.

[Patil 81] Patil, R.S.
Causal Representation of Patient Illness for Electrolyte and Acid-Base Diagnosis.
PhD thesis, Department of Computer Science, M.I.T., October, 1981.
Report no. LCS-TR-267.

[Pauker 76] Pauker, S.G., Gorry, G.A., Kassirer, J.P., and Schwartz, W.B.
Toward the simulation of clinical cognition: Taking a present illness by computer.
American Journal of Medicine 60:981-995, 1976.

[Pearl 82] Pearl, J.
Reverend Bayes on inference engines: A distributed hierarchical approach.
In *Proceedings of the Second National Conference on Artificial Intelligence*, pages 133-136. The American Association for Artificial Intelligence, 1982.

[Perlroth 81] Perlroth, M.G., and Weiland, D.J.
Fifty Diseases: Fifty Diagnoses.
Year Book Medical Publisher, Chicago, Illinois, 1981.

[Petersdorf 83] Petersdorf, R.G., et. al. (editors).
Harrison's Principles of Internal Medicine.
McGraw-Hill, New York, 1983.

[Pople 75] Pople, H.E., Jr., Myers, J.D., and Miller, R.A.
DIALOG: A model of diagnostic logic for internal medicine.
In *Proceedings of the Fourth International Joint Conference on Artificial Intelligence.* IJCAI, 1975.

[Pople 77] Pople, H.E., Jr.
The formation of composite hypotheses in diagnostic problem-solving: An exercise in synthetic reasoning.
In *Proceedings of the Fifth International Joint Conference on Artificial Intelligence.* IJCAI, 1977.

[Pople 82] Pople, H.E., Jr.
Heuristic methods for imposing structure on ill-structured problems: The structuring of medical diagnostics.
In Szolovits, P. (editor), *Artificial Intelligence in Medicine*, . Westview Press, Inc., Boulder, Colorado, 1982.

[Rousseau 68] Rousseau, W.F.
A method for computing probabilities in complex situations.
Technical Report 6252-2, Stanford University Center for Systems Research, May, 1968.

[Rowe 83] Rowe, N.C.
Rule-Based Statistical Calculations on a Database Abstract.
PhD thesis, Department of Computer Science, Stanford University, 1983.

[Schwartz 70] Schwartz, W.B.
Medicine and the computer: The promise and problems of change.
The New England Journal of Medicine 283:1257-1264, 1970.

[Schwartz 73] Schwartz, W.B., Gorry, G.A., Kassirer, J.P., and Essig, A.
Decision analysis and clinical judgement.
The American Journal of Medicine 55:459-472, 1973.

[Shafer 76] Shafer, G.
A Mathematical Theory of Evidence.
Princeton University Press, Princeton, NJ, 1976.

[Shortliffe 74] Shortliffe, E.H.
MYCIN: A Rule-Based Computer Program for Advising Physicians Regarding Antimicrobial Therapy Selection.
PhD thesis, Stanford University, 1974.

[Shortliffe 75] Shortliffe, E.H., and Davis, R.
Some considerations for the implementation of knowledge-based expert systems.
SIGART Newsletter 55:9-12, 1975.

[Shortliffe 76] Shortliffe, E.H.
Computer-Based Medical Consultations: MYCIN.
Elsevier, New York, 1976.

[Shortliffe 79] Shortliffe, E.H., Buchanan, B.G., and Feigenbaum, E.A.
Knowledge engineering for medical decision making: A review of computer-based clinical decision aids.
Proceedings of the IEEE 67:1207-1224, 1979.

[Spiegelhalter 84] Spiegelhalter, D.J., and Knill-Jones, R.P.
Statistical and knowledge-based approaches to clinical decision support systems, with an application in gastroenterology.
Journal of the Royal Statistical Society 147:35-77, 1984.

[Swartout 81] Swartout, W.R.
Producing Explanations and Justifications of Expert Consulting Programs.
PhD thesis, Department of Computer Science, M.I.T., 1981.
Report no. LCS-TR-251.

[Swinburne 73] Swinburne, R.G.
An Introduction to Confirmation Theory.
Methuen & Co., Ltd., London, 1973.

[Szolovits 78] Szolovits, P., and Pauker, S.G.
Categorical and probabilistic reasoning in medical diagnosis.
Artificial Intelligence 11:115-144, 1978.

[Tversky 74] Tversky, A., and Kahneman, D.
Judgement under uncertainty: Heuristics and biases.
Science 185:1124-1131, 1974.

[vanMelle 80] van Melle, W.
A Domain-Independent System that Aids in Constructing Knowledge-Based Consultation Programs.
PhD thesis, Department of Computer Science, Stanford University, 1980.

[Wagner 78] Wagner, G., Tauta, P., and Wolber, U.
Problems of medical diagnosis - a bibliography.
Methods of Information in Medicine 17:55-74, 1978.

[Wallis 82] Wallis, J.W., and Shortliffe, E.H.
Explanatory power for medical expert systems: Studies in the representation of causal relationships for clinical consultation.
Methods of Information in Medicine 21:127-136, 1982.

[Warner 61] Warner, H.R., Toronto, A.F., Veasy, L.G., and Stephenson, R.
A mathematical approach to medical diagnosis: Application to congenital heart disease.
Journal of the American Medical Association 177:177-183, 1961.

[Weiner 80] Weiner, J.L.
BLAH: A system which explains its reasoning.
Artificial Intelligence 15:19-48, 1980.

[Weinstein 80] Weinstein, M.C., Fineberg, H.V., et.al.
Clinical Decision Analysis.
W.B. Saunders. Philadelphia, PA, 1980.

[Weiss 78] Weiss, J.M., Kulikowski, C.A., Amarel, S., and Safir, A.
A model-based method for computer-aided medical decision-making.
Artificial Intelligence 11:145-172, 1978.

[Winston 84] Winston, P.H.
Artificial Intelligence.
Addison-Wesley, Reading, Massachusetts, 1984.

[Yu 79a] Yu, V.L., Buchanan, B.G., Shortliffe, E.H., Wraith, S.M., Davis, R., Scott, A.C., and Cohen, S.N.
An evaluation of the performance of a computer-based consultant.
Computer Programs in Biomedicine 9:95-102, 1979.

[Yu 79b] Yu, V.L., Fagan, L.M., Wraith, S.M., Clancey, W.J., Scott, A.C., Hannigan, J.F., Blum, R.L., Buchanan, B.G., and Cohen, S.N.
Antimicrobial selection by computer: A blinded evaluation by infectious disease experts.
Journal of the American Medical Association 242:1279-1282, 1979.

Distribution Statement

Defense Documentation Center Cameron Station Alexandria, VA 22314	12 copies
Office of Naval Research Arlington, VA 22217	
Information Systems Program (437)	2 copies
Code 200	1 copy
Code 455	1 copy
Code 458	1 copy
Office of Naval Research Eastern/Central Regional Office Bldg. 114 Section D 666 Summer St. Boston, MA 02210	1 copy
Office of Naval Research Branch Office, Chicago 536 South Clark St. Chicago, IL 60605	1 copy
Office of Naval Research Western Regional Office 1030 East Green St. Pasadena, CA 91106	1 copy
Naval Research Laboratory Technical Information Division, Code 2627 Washington, DC 20375	6 copies
Dr. A. L. Slafkosky Scientific Advisor Commandant of the Marine Corps (RD-1) Washington, DC 20380	1 copy
Naval Ocean Systems Center Advanced Software Technology Division Code 5200 San Diego, CA 92152	1 copy
Mr. E. H. Gleissner Naval Ship Research & Development Center Computation and Mathematics Department Bethesda, MD 20084	1 copy
Capt. Grace M. Hopper (008) Naval Data Automation Command Washington Navy Yard Bldg. 166 Washington, DC 20374	1 copy
Captain R. Martin 3311 West Avenue Newport News, VA 23607	1 copy

ONR Resident Representative
Durand Aeronautics Bldg., Rm. 165
Stanford University
Stanford, California 94305

1 copy

Edward H. Shortliffe, M.D., Ph.D.
Stanford Med Ctr TC-117

1 copy

Michael R. Genesereth, Ph.D.
Computer Science Department
Stanford University

1 copy

END

FILMED

5-85

DTIC